TEMPO: FEATURE-ENDOWED TEICHMÜLLER EXTREMAL MAPPINGS OF POINT CLOUDS

TING WEI MENG, PUI TUNG CHOI AND LOK MING LUI

Abstract. In recent decades, the use of 3D point clouds has been widespread in computer industry. The development of techniques in analyzing point clouds is increasingly important. In particular, mapping of point clouds has been a challenging problem. In this paper, we develop a discrete analogue of the Teichmüller extremal mappings, which guarantee uniform conformality distortions, on point cloud surfaces. Based on the discrete analogue, we propose a novel method called TEMPO for computing Teichmüller extremal mappings between feature-endowed point clouds. Using our proposed method, the Teichmüller metric is introduced for evaluating the dissimilarity of point clouds. Consequently, our algorithm enables accurate recognitions and classifications of point clouds. Experimental results demonstrate the effectiveness of our proposed method.

Key words. Teichmüller extremal mapping, Teichmüller metric, point clouds, shape analysis, face recognition, quasi-conformal theory

1. Introduction. In recent years, the use of point clouds has been widespread for a large variety of industrial applications. In this emerging field, one important topic is point cloud mapping, which has a wide range of applications such as 3D facial recognition and object classifications. The difficulty of the mapping problem arises from the limited information a point set provides. Unlike mesh representations of surfaces, point clouds do not contain any information about the connectivity structure. This hinders the extension of the well-established algorithms on surface meshes to point clouds.

In particular, accurate registrations between two feature-endowed point clouds are challenging because of the enforcement of feature correspondences. Besides satisfying the feature correspondences, an admissible mapping between two feature-endowed point clouds should also retain the underlying geometric structures as complete as possible. In differential geometry, conformal mappings are angle preserving and hence preserve the local geometries. Note that under general feature correspondences, conformal mappings between two surfaces may not exist. Nevertheless, we can consider quasi-conformal mappings, the extension of conformal mappings. Among all quasi-conformal mappings, a special class of landmark matching surface diffeomorphisms, called the *Teichmüller mappings (T-maps)*, are particularly desirable as they ensure uniform conformality distortions. In other words, Teichmüller mappings effectively achieve evenly distributed geometric distortions under prescribed feature point correspondences.

In [32], Lui et al. proposed an efficient method for computing the Teichmüller mapping on triangular meshes using the Beltrami coefficients. In this paper, we explore Teichmüller mappings on point clouds in both the theoretical and the computational aspects. A novel discrete analogue of Teichmüller mappings on point clouds is rigorously developed. Then, we extend and modify the algorithm in [32] for point clouds with underlying surfaces being simply-connected open surfaces. More specifically, we first introduce a hybrid quasi-conformal reconstruction scheme on point clouds with given Beltrami coefficients. Then, we propose an improved method for approximating differential operators arisen from the computation of conformal and quasi-conformal mappings on point clouds with disk topology. With the above-mentioned schemes, we propose an efficient method for computing the **T**eichmüller **E**xtremal **M**apping of **PO**int clouds, abbreviated as TEMPO. To register two disk-

type point clouds with given feature correspondences, our TEMPO algorithm first parameterizes them onto two rectangular regions on the complex plane using conformal mappings. Then, the Teichmüller mapping between the two rectangular regions with specific feature correspondences is computed. The Teichmüller parameterization naturally induces a metric, called the *Teichmüller metric*, to assess the dissimilarity of two feature-endowed point clouds. Hence, the Teichmüller metric provides us with an accurate way to classify point clouds. Experimental results are presented to demonstrate the effectiveness of our proposed scheme for feature-endowed disk-type point clouds.

The organization of this paper is outlined as follows. In Section 2, we review the previous works on parameterizations of simply-connected surfaces and point clouds. The contributions of our work are highlighted in Section 3. In Section 4, we introduce some mathematical concepts related to our work. In Section 5, we develop a novel discrete analogue of the continuous Teichmüller mappings on point clouds. In Section 6, we explain our proposed TEMPO algorithm for computing Teichmüller mappings between feature-endowed point clouds in details. Experimental results are reported in Section 7 to illustrate the effectiveness of our proposed method. The paper is concluded in Section 8.

2. Previous Works. In this section, we review the previous works on conformal and quasi-conformal parameterizations of disk-type meshes and point clouds.

In the past few decades, extensive studies on mesh parameterization have been carried out by various research groups. Surveys of mesh parameterization methods can be found in [10, 12, 19, 43]. In [7], Floater introduced the shape-preserving mesh parameterization method, which involves solving linear systems based on convex combinations. Hormann and Greiner [18] presented the Most Isometric Parameterization of Surfaces method (MIPS) for disk-like surface meshes. Sheffer and de Sturler [41] proposed the Angle Based Flattening (ABF) method for mesh parameterizations. The method solves the parameterization problem as a constrained optimization problem in terms of angles only. Later, Sheffer et al. [42] extended the ABF method to the ABF++ method. A new numerical solution technique and an efficient hierarchical technique are introduced to overcome the drawbacks of the ABF method. In [26], Lévy et al. proposed the Least Square Conformal Maps (LSCM). The parameterization method is based on a least-square approximation of the Cauchy-Riemann equations. Desbrun et al. [6] proposed the Discrete, Natural Conformal Parameterization (DNCP) by computing the discrete Dirichlet energy without fixing the boundary positions. The LSCM and the DNCP were later shown to be equivalent. Motivated by the Mean Value Theorem for harmonic functions, Floater [11] derived a generalization of barycentric coordinates for planar parameterizations. Kharevych et al. [23] introduced a method for constructing conformal parameterizations of meshes based on circle patterns, which are arrangements of circles with prescribed intersection angles. In [14], Gu and Yau computed conformal parameterizations by constructing a basis of holomorphic 1-forms and integrating holomorphic differentials. In [34], Luo proposed the combinatorial Yamabe flow for the parameterization of triangulated surfaces. Jin et al. [21] computed disk conformal parameterizations using the double covering technique [15] and the spherical conformal mapping algorithm [16]. In [38], Mullen et al. presented a spectral approach to compute free-boundary conformal parameterizations of triangular meshes. The approach involves solving a sparse symmetric generalized eigenvalue problem. In [22], Jin et al. proposed the discrete surface Ricci flow algorithm for conformal parameterizations of triangular meshes. The discrete curvature

determines the deformation of edge lengths of the triangles and hence the discrete Ricci flow. Later, Yang et al. [48] generalized the traditional discrete Ricci flow by allowing the two circles either intersect or separate from each other in circle packing. In [50], Zhang et al. introduced the unified theoretic framework for discrete surface Ricci flow. The unified framework improved the robustness and efficiency of conformal parameterizations. In [3], Choi and Lui proposed a two-step iterative scheme for computing disk conformal parameterizations of simply-connected open meshes. The conformality distortion of the parameterization is corrected in two steps using quasi-conformal theories. Recently, Choi and Lui [4] introduced a linear algorithm for disk conformal parameterizations, with the aid of a linear spherical conformal parameterization algorithm [2].

In recent years, several methods for quasi-conformal parameterizations of surface meshes have been developed. Weber et al. [44] presented an algorithm for computing extremal quasi-conformal mappings for simply-connected open meshes using holomorphic quadratic differentials. In [29], Lipman introduced convex mapping spaces for triangular meshes with guarantees on the maximal conformal distortion and injectivity of their mappings. The methods were applied to compute quasi-conformal parameterizations. In [32], Lui et al. proposed a simple iterative algorithm for computing Teichmüller mappings of simply-connected open meshes with the aid of Beltrami coefficients. The convergence of the algorithm was recently proved in [33].

With the increasing popularity of point clouds, numerous methods on parameterizing and registering point clouds have been developed. When compared with the case of meshes, parameterizing point clouds with disk topology is a more challenging task because of the absence of the connectivity information. Floater and Reimers [8, 9] presented a meshless parameterization method by solving a sparse linear system. In [17], Guo et al. used Riemann surface theory and Hodge theory to obtain point cloud conformal parameterizations. Zhang et al. [49] proposed an as-rigid-as-possible (ARAP) parameterization method for point clouds with disk topology. Liang et al. [27] computed the spherical conformal parameterizations of genus-0 point clouds by approximating the Laplace-Beltrami operator. Choi et al. [5] proposed the North-South reiterations for computing the spherical conformal parameterizations of genus-0 point clouds. The method can also be extended to obtain planar conformal parameterizations of disk-type point clouds. In [35], Meng and Lui computed quasi-conformal parameterizations of disk-type point clouds by developing the approximation theories of quasi-conformal maps and Beltrami coefficients on point clouds. Table 2.1 lists different parameterization schemes for point clouds with disk topology.

3. Contribution. In this work, we are concerned with both the theory and the computation of Teichmüller extremal mappings on point clouds. To the best of our knowledge, this is the first work on the Teichmüller mappings on point clouds.

For the theoretical aspect, we develop the concept of PCT-maps, a novel discrete analogue of the continuous Teichmüller mappings on point clouds. Rigorous theories are established to show the relationship between the discrete PCT-maps and the continuous Teichmüller mappings.

For the computational aspect, our proposed TEMPO method for computing the Teichmüller mappings between feature-endowed point clouds with disk topology is advantageous for the following reasons:

    1. We introduce a new hybrid quasi-conformal reconstruction scheme for point clouds with given Beltrami coefficients. The scheme enables us to extend the existing mesh-based methods for computing quasi-conformal mappings

Methods	Criteria for the distortion	Landmark constraints allowed?
Meshless parameterization [8, 9]	/	No
Point-based global parameterization [17]	Conformal except a number of zero points	No
As-rigid-as-possible meshless parameterization [49]	As-rigid-as-possible	No
North-South reiteration [5]	Conformal	No
Quasi-conformal parameterization [35]	Quasi-conformal	No
Our proposed TEMPO method	Teichmüller (uniform conformality distortion)	Yes

TABLE 2.1
A comparison on the parameterization schemes for point clouds with disk topology.

to point cloud surfaces.

2. We introduce an improved method for approximating differential operators arisen from the computation of conformal and quasi-conformal mappings on point clouds with disk topology by combining the Moving Least Square (MLS) method in [27, 28, 5] and the local mesh method in [24].

3. Unlike the existing approaches for point cloud parameterizations, our proposed Teichmüller parameterization method allows landmark constraints.

4. As Teichmüller mappings ensure uniform conformality distortions, our method provides a natural registration result without uneven distortions.

5. Our Teichmüller parameterization method is highly efficient. The computation typically completes within 1 minute for moderate point clouds, and requires only a few minutes even for very dense point clouds.

6. Using the induced Teichmüller metric, accurate classifications of point clouds can be easily achieved.

7. Besides the computational advantages, our TEMPO method is theoretically supported. The limit mapping obtained by our algorithm is proved to be a PCT-map.

4. Mathematical background. In this section, we introduce some mathematical concepts closely related to our work. Readers are referred to [40, 13, 32] for more details.

4.1. Conformal mappings. First, we introduce the concept of conformal mappings.

Definition 4.1 (Conformal mappings). *Let M and N be two Riemann surfaces. A mapping $f : M \to N$ is said to be a* conformal mapping *if there exists a positive scalar function λ such that*

$$f^* ds_{\mathcal{N}}^2 = \lambda ds_{\mathcal{M}}^2. \tag{4.1}$$

The mentioned scalar function λ is called the *conformal factor*. It follows that every conformal mapping preserves angles and hence the local geometry of the surface
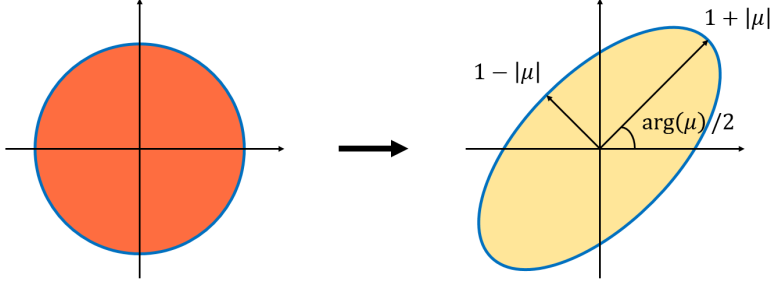
FIG. 4.1. *An geometric illustration of quasi-conformal mappings. The information about the maximal magnification and shrinkage can be determined by the associated Beltrami coefficient μ.*

structures. By the uniformization theorem, every simply-connected Riemann surface is conformally equivalent to one of the following domains:

1. the open unit disk,
2. the complex plane, and
3. the Riemann sphere.

In this work, we focus on point clouds sampled from simply-connected open surfaces. With the theoretical guarantee by the uniformization theorem, it is natural to consider conformally parameterizing the point clouds onto simple planar domains, such as the open unit disk or a rectangle, for further analyses. However, with the presence of feature point correspondences, conformal mappings between two surfaces may not exist. In this case, we need to introduce a generalization of conformal mappings.

4.2. Quasi-conformal mappings. A generalization of conformal mappings is the quasi-conformal mappings, which are orientation preserving homeomorphisms between Riemann surfaces with bounded conformality distortion. Intuitively, conformal mappings map infinitesimal circles to infinitesimal circles, while quasi-conformal mappings map infinitesimal circles to infinitesimal ellipses with bounded eccentricity [13]. Mathematically, quasi-conformal mappings can be defined as follows.

Definition 4.2 (Quasi-conformal mappings). *$f : \mathbb{C} \to \mathbb{C}$ is a quasi-conformal mapping if it satisfies the Beltrami equation*

$$\frac{\partial f}{\partial \overline{z}} = \mu(z) \frac{\partial f}{\partial z} \tag{4.2}$$

for some complex-valued function μ satisfying $||\mu||_\infty < 1$ and that $\frac{\partial f}{\partial z}$ is non-vanishing almost everywhere.

The complex-valued function μ is called the *Beltrami coefficient* of the quasi-conformal mapping f. The Beltrami coefficient serves as a measure of the non-conformality of f. It measures how far the mapping f at each point is deviated from a conformal mapping. A geometric illustration of quasi-conformal mappings is provided in Figure 4.1. In addition, the *maximal quasi-conformal dilation* of f is given by

$$K(f) = \frac{1 + ||\mu||_\infty}{1 - ||\mu||_\infty}. \tag{4.3}$$

For the composition of two quasi-conformal mappings, the associated Beltrami coefficient can be expressed in terms of the Beltrami coefficients of the two mappings.

THEOREM 4.1. *Let $f : \Omega \subset \mathbb{C} \to f(\Omega)$ and $g : f(\Omega) \to \mathbb{C}$ be two quasi-conformal mappings. Then the Beltrami coefficient of $g \circ f$ is given by*

$$\mu_{g \circ f} = \frac{\mu_f + (\overline{f_z}/f_z)(\mu_g \circ f)}{1 + (\overline{f_z}/f_z)\overline{\mu_f}(\mu_g \circ f)}. \tag{4.4}$$

Conversely, a quasi-conformal mapping can also be determined by a complex function. Given a Beltrami coefficient $\mu : \mathbb{C} \to \mathbb{C}$ with $\|\mu\|_\infty < 1$, there is always a quasi-conformal mapping from $\mathbb{C}$ onto itself which satisfies the Beltrami equation (4.2) in the distribution sense [13]. Therefore, finding an optimal quasi-conformal mapping is equivalent to finding an optimal complex-valued function.

Besides, given a quasi-conformal mapping $(f, |dz|^2)$ associated with the Beltrami coefficient μ, we can consider the distorted metric tensor $|dz + \mu d\bar{z}|^2$. Then, the mapping $(f, |dz + \mu d\bar{z}|^2)$ is in fact a conformal mapping with respect to the distorted metric.

4.3. Teichmüller mappings. Recall that quasi-conformal mappings are with bounded conformality distortions. In the space of all quasi-conformal mappings, it is natural to consider a type of quasi-conformal mappings, called the Teichmüller mappings, that produce uniform conformality distortions with prescribed landmark constraints. Mathematically, the Teichmüller mappings are defined as follows.

Definition 4.3 (Teichmüller mapping). *Let $f : M \to N$ be a quasi-conformal mapping. f is said to be a* Teichmüller mapping (T-map) *associated with the quadratic differential $q = \varphi dz^2$ where $\varphi : M \to \mathbb{C}$ is a holomorphic function if its associated Beltrami coefficient is of the form*

$$\mu(f) = k\frac{\overline{\varphi}}{|\varphi|} \tag{4.5}$$

for some constant $k < 1$ and quadratic differential $q \neq 0$ with $||q||_1 = \int_{S_1} |\varphi| < \infty$.

It follows that a Teichmüller mapping is a quasi-conformal mapping whose Beltrami coefficient has a constant norm. Hence, a Teichmüller mapping has a uniform conformality distortion over the whole domain.

Furthermore, Teichmüller mappings are closely related to a class of mappings called extremal quasi-conformal mappings. Before explaining their relationships, we introduce the precise definition of extremal quasi-conformal mappings.

Definition 4.4 (Extremal quasi-conformal mapping). *Let $f : M \to N$ be a quasi-conformal mapping. f is said to be an* extremal quasi-conformal mapping *if for any quasi-conformal mapping $h : M \to N$ isotopic to f relative to the boundary, we have*

$$K(f) \leq K(h) \tag{4.6}$$

where $K(f)$ is the maximal quasi-conformal dilation of f. It is uniquely extremal if the inequality (4.6) is strict when $h \neq f$.

Under suitable boundary conditions, the Teichmüller mappings and the extremal mappings from the unit disk to the unit disk are in fact equivalent.

Theorem 4.5 (Landmark-matching Teichmüller mapping of D). *Let $g : \partial D \to \partial D$ be an orientation-preserving diffeomorphism of ∂D, where D is the unit disk. Suppose further that $g'(e^{i\theta}) \neq 0$ and $g''(e^{i\theta})$ is bounded. Let $\{p_i\}_{i=1}^n \in D$ and $\{q_i\}_{i=1}^n \in D$*

be the corresponding interior landmark constraints. Then there is a Teichmüller mapping $f : (D, \{p_i\}_{i=1}^n) \to (D, \{q_i\}_{i=1}^n)$ matching the interior landmarks, which is the unique extremal extension of g to D.

Therefore, besides equipped with uniform conformality distortions, Teichmüller mappings are extremal in the sense that they minimize the maximal quasi-conformal dilation K. In this paper, our goal is to compute landmark-matching Teichmüller extremal mappings between point clouds. For more details, please refer to [32].

4.4. Point clouds and some related concepts. In this subsection, we briefly introduce point clouds and some mathematical concepts about this data structure. To avoid ambiguity, normal characters are used for introducing the continuous theories and bold characters are used for the mappings of point clouds.

Intuitively, a *point cloud* $P = \{p_1, p_2, \ldots, p_N\} \subset \mathbb{R}^3$ is a set of points sampled from a Riemann surface $\mathcal{M}$. $\mathcal{M}$ is called the underlying surface of P. In this paper, we focus on disk-like point clouds, the point clouds with the underlying surfaces being simply-connected open surfaces. As disk-like point clouds can be conformally mapped onto a rectangular domain, it suffices to introduce the concepts on planar point clouds on $\mathbb{R}^2$.

A *planar point cloud* $P = \{p_1, p_2, \ldots, p_N\} \subset \Omega$ is a set of points sampled from a disk-type domain $\Omega \subset \mathbb{R}^2$, where both the domains and the sampled point clouds are assumed to satisfy the following conditions [47].

Definition 4.6 (Interior cone condition). *A domain $\Omega \subset \mathbb{R}^2$ is said to* satisfy an interior cone condition *with parameter $r > 0$ and $\theta \in (0, \pi/2)$ if for each $x \in \Omega$, there exists a unit vector $d(x)$ such that $C(x, d, \theta, r) \subseteq \Omega$, where*

$$C(x, d, \theta, r) = \{x + ty : y \in \mathbb{S}^1, \, y^T d(x) \geq \cos\theta, \, 0 \leq t \leq r\}. \tag{4.7}$$

Definition 4.7 (Quasi-uniform point cloud). *Let P be a point cloud sampled from a planar domain $\Omega \subset \mathbb{R}^2$. The* fill distance $h_{P,\Omega}$ *and the* separation distance q_P *are respectively defined by*

$$h_{P,\Omega} := \sup_{x \in \Omega} \min_{p \in P} \|x - p\| \tag{4.8}$$

and

$$q_P := \frac{1}{2} \min_{\substack{p_1, p_2 \in P \\ p_1 \neq p_2}} \|p_1 - p_2\|. \tag{4.9}$$

P is said to be quasi-uniform *with positive constant c_{qu} if*

$$q_P \leq h_{P,\Omega} \leq c_{qu} q_P. \tag{4.10}$$

From now on, we assume that all planar point clouds are quasi-uniform and are sampled from domains which satisfy an interior cone condition. On point clouds, it is difficult to compute interpolation functions. In order to compute quasi-conformal mappings on point clouds, it is important to develop a suitable approximation scheme.

One common approach to approximate functions and derivatives on point clouds is the *Moving Least Square (MLS)* method [30, 25, 39, 1, 27, 28, 5]. A concise introduction of the second order MLS method is given below. For more details, we refer the readers to [47, 36, 37].

Weight	Formula of $w(d)$
Constant weight	$w(d) = 1$
Exponential weight	$w(d) = \exp\left(-\dfrac{d^2}{D^2}\right)$
Inverse of squared distance weight	$w(d) = \dfrac{1}{d^2 + \epsilon^2}$
Wendland weight [45, 46]	$w(d) = \left(1 - \dfrac{d}{D}\right)^4 \left(\dfrac{4d}{D} + 1\right)$
Special weight [27]	$w(d) = \begin{cases} 1 & \text{if } d = 0 \\ \frac{1}{K} & \text{if } d \neq 0 \end{cases}$
Gaussian weight [5]	$w(d) = \begin{cases} 1 & \text{if } d = 0 \\ \frac{1}{K}\exp(\frac{-\sqrt{K}d^2}{D^2}) & \text{if } d \neq 0 \end{cases}$

TABLE 4.1

Some common weight functions for the moving least square approximation. Here K is the number of points in the chosen neighborhood, and D is the maximal distance of the neighborhood.

Given a function $f : \Omega \to \mathbb{R}$ and a disk-type point cloud P sampled from Ω with fill distance h, the MLS method approximates f locally near every point. Define the mappings $q, q_1, q_2 : \Omega \to \mathbb{R}^6$ respectively by

$$q(x_1, x_2) = [1, x_1, x_2, x_1^2, x_1 x_2, x_2^2]^T, \tag{4.11}$$

$$q_1(x_1, x_2) = \partial_1 q = [0, 1, 0, 2x_1, x_2, 0]^T, \tag{4.12}$$

and

$$q_2(x_1, x_2) = \partial_2 q = [0, 0, 1, 0, x_1, 2x_2]^T. \tag{4.13}$$

Then, the local approximation near x is given by $c_x^T q(y)$, where

$$c_x = argmin_c \sum_{i=1}^{N} w(\|x - p_i\|) \left(c^T q(p_i) - f_i\right)^2. \tag{4.14}$$

Here w is a weight function compactly supported in $[0, C_\delta h]$ and C_δ is a constant defined in [47]. The solution c_x can be expressed as

$$c_x = (Q^T W(x) Q)^{-1} Q^T W(x) F, \tag{4.15}$$

where $W(x)$ is a diagonal matrix whose diagonal elements are $W_{ii} = w(\|x - p_i\|)$, and Q, F are respectively the matrix forms of q and f. From now on, unless otherwise specified, we use the corresponding capital letter of a given point cloud mapping to denote its matrix form, in which the i-th row is the function value on p_i.

Numerous studies have been devoted to the choice of the weight function w in the MLS method. Some common weight functions are listed in Table 4.1. Denote

$$A_x = (Q^T W(x) Q)^{-1} Q^T W(x). \tag{4.16}$$

Then, the function is locally approximated by

$$\tilde{f}(y) = q^T(y) A_x F. \tag{4.17}$$

Hence, the derivative approximation naturally follows. In Section 6, we establish a novel approximation scheme based on the MLS method to compute quasi-conformal map on planar point clouds.

Besides, recall that in quasi-conformal theory, quasi-conformal mappings are closely related to the Beltrami coefficients. Therefore, to compute quasi-conformal mappings between point clouds, it is necessary to extend the definition of the Beltrami coefficients to point clouds. In this work, we adopt the definition of *discrete diffuse point cloud Beltrami coefficients (PCBC)* proposed by Meng and Lui in [35] for the computation of the Beltrami coefficient $\tilde{\boldsymbol{\mu}}$ on point clouds.

Definition 4.8 (Discrete diffuse point cloud Beltrami coefficients [35]). *Given a point cloud $P = \{p_i \in \mathbb{R}^2 : i = 1, \cdots, N\}$ with underlying domain Ω and a target point cloud function $\mathbf{f} : P \to \mathbb{R}^2$, where $\mathbf{f} = (\mathbf{u}, \mathbf{v})^T$, the diffuse point cloud Beltrami coefficient (PCBC) $\tilde{\mu} : \Omega \to \mathbb{C}$ is defined by*

$$\tilde{\mu}(x) = \frac{\begin{bmatrix} q_1^T(x) & q_2^T(x) \end{bmatrix} \begin{bmatrix} A_x & iA_x \\ iA_x & -A_x \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix}}{\begin{bmatrix} q_1^T(x) & q_2^T(x) \end{bmatrix} \begin{bmatrix} A_x & iA_x \\ -iA_x & A_x \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix}}, \tag{4.18}$$

where q_1, q_2 and A_x are respectively defined as in Equation (4.12), Equation (4.13) and Equation (4.16). In addition, the discrete diffuse PCBC is a complex valued point cloud function defined by $\tilde{\boldsymbol{\mu}} = \tilde{\mu}|_P$.

With the abovementioned definition of the Beltrami coefficients for point clouds, we are ready to introduce our proposed TEMPO method for computing landmark-endowed Teichmüller extremal mappings between point clouds.

5. A discrete analogue of Teichmüller mappings on point clouds. The continuous Teichmüller mappings have been extensively studied by mathematicians. By contrast, the development of the Teichmüller mappings on point clouds remains a great challenge. In this section, we establish a novel discrete version of the Teichmüller extremal mappings on point clouds with theoretical guarantees. First, we define the discrete version of Teichmüller mappings on planar point clouds.

Definition 5.1 (Planar point cloud Teichmüller mappings). *Let P be a quasi-uniform point cloud with underlying domain $\Omega \subset \mathbb{R}^2$ satisfying the interior cone condition. A point cloud mapping $\mathbf{f}$ is called a planar point cloud Teichmüller mapping (planar PCT-map) if its discrete diffuse PCBC $\boldsymbol{\mu}$ has a constant norm $k = |\boldsymbol{\mu}(p)|$ for any $p \in P$, and satisfies the following equation*

$$(L_i\boldsymbol{\mu})\overline{\boldsymbol{\mu}(p_i)} \in \mathbb{R} \tag{5.1}$$

for any interior point $p_i \in P$. Here L is the approximation of the Laplace-Beltrami operator by the MLS method, and L_i denotes the i-th row of L.

Our goal is to prove that under certain convergence conditions, the planar PCT-maps converge to the continuous T-maps defined on the underlying domain. We now consider the following lemma.

LEMMA 5.1. *Let R be any rectangular domain, and $u, v \in C^3(R)$ satisfy $u\Delta v = v\Delta u$ and $u^2 + v^2 = 1$. Then there exists a real valued function f such that $\phi = f \cdot (u + iv)$ is holomorphic, and $|f| \neq 0$ in the domain.*

Proof. Consider the Cauchy-Riemann equation

$$
\begin{aligned}
f_x u + u_x f = f_y v + v_y f; \\
f_y u + u_y f = -f_x v - v_x f.
\end{aligned}
\tag{5.2}
$$

Since $u^2 + v^2 = 1$, the above system is equivalent to

$$
\begin{aligned}
f_x = G_1 f; \\
f_y = G_2 f,
\end{aligned}
\tag{5.3}
$$

where

$$
\begin{aligned}
G_1 = -uu_x - vv_x - vu_y + uv_y; \\
G_2 = -uu_y - vv_y - uv_x + vu_x.
\end{aligned}
\tag{5.4}
$$

Let the domain R be $[a_1, a_2] \times [b_1, b_2]$. By solving the first equation above, we obtain

$$
f = \exp\left(\int_{a_1}^x G_1(s, y)ds + g(y)\right),
\tag{5.5}
$$

where g is a function of y. Plugging this solution to the second equation, we have

$$
\int_{a_1}^x \partial_y G_1(s, y)ds + g_y(y) = G_2.
\tag{5.6}
$$

This equation has a solution if $\partial_y G_1 = \partial_x G_2$. By direct calculation,

$$
\partial_y G_1 - \partial_x G_2 = u\Delta v - v\Delta u = 0.
\tag{5.7}
$$

Therefore, this system has a solution in the form of Equation (5.5), where

$$
g = \int_{b_1}^y G_2(x, t)dt - \int_{a_1}^x (G_1(s, y) - G_1(s, b_1))ds + C.
\tag{5.8}
$$

Consequently, the solution f is

$$
f = \exp\left(\int_{b_1}^y G_2(x, t)dt + \int_{a_1}^x G_1(s, b_1)ds + C\right).
\tag{5.9}
$$

It is obvious that $|f| \neq 0$. Moreover, as $f \cdot (u + iv)$ is a continuous function satisfying the Cauchy-Riemann equation, it is holomorphic. This proves the lemma. $\square$

With the aid of the above lemma, we are ready to establish the following theorem about the consistency between planar PCT-maps and the continuous T-maps.

Proposition 5.2. *Let $\{P_n\}$ be an ascending sequence of point clouds (in other words, $P_1 \subset P_2 \subset \cdots P_n \subset P_{n+1} \subset \cdots$) sampled from a rectangular domain R_1 with fill distance $h_n \to 0$. For each P_n, let $\mathbf{g}_n$ be a PCT-map defined on it with discrete diffuse PCBC $\tilde{\boldsymbol{\mu}}_n$. Further assume that there exists a smooth quasi-conformal mapping $g : R_1 \to R_2$ with Beltrami coefficient μ such that for any point $p \in P := \cup_m P_m$,*

$\mathbf{g}_n(p)$ *converges to $g(p)$. Let the sup-norm of the error be given by $\epsilon_n = \|\mathbf{g}_n - g|_{P_n}\|_\infty$ and assume that $\lim_n \epsilon_n h_n^{-2} = 0$. Then g is a Teichmüller mapping.*

Proof. First, we prove that $\tilde{\boldsymbol{\mu}}_n(p)$ converges to $\mu(p)$ for each $p \in P_m$, where m is an arbitrary number.

Let $\tilde{\boldsymbol{\nu}}_n$ be the discrete diffuse PCBC of a point cloud mapping $g|_{P_n}$. We have

$$|\mu(p) - \tilde{\boldsymbol{\mu}}_n(p)| \le |\mu(p) - \tilde{\boldsymbol{\nu}}_n(p)| + |\tilde{\boldsymbol{\nu}}_n(p) - \tilde{\boldsymbol{\mu}}_n(p)|. \tag{5.10}$$

It is proved in [35] that $|\mu(p) - \tilde{\boldsymbol{\nu}}_n(p)| \le C(g)h_n^2$. Let $\tilde{G}$ and $\tilde{G}_n$ be the vector forms of the point cloud mappings $g|_{P_n}$ and $\mathbf{g}_n$. For simplicity, we adopt the following notation.

$$
\begin{aligned}
D_1 &= \begin{pmatrix} q_1^T(x) & q_2^T(x) \end{pmatrix} \begin{pmatrix} A_x & iA_x \\ iA_x & -A_x \end{pmatrix}, \\
D_2 &= \begin{pmatrix} q_1^T(x) & q_2^T(x) \end{pmatrix} \begin{pmatrix} A_x & iA_x \\ -iA_x & A_x \end{pmatrix}.
\end{aligned}
\tag{5.11}
$$

Then we have

$$
\begin{aligned}
&|\tilde{\boldsymbol{\nu}}_n(p) - \tilde{\boldsymbol{\mu}}_n(p)| \\
&\le \left| \frac{D_1(p)\tilde{G}}{D_2(p)\tilde{G}} - \frac{D_1(p)\tilde{G}_n}{D_2(p)\tilde{G}_n} \right| \\
&\le \frac{\left| D_1(p)\tilde{G}D_2(p)\left(\tilde{G}_n - \tilde{G}\right) \right| + \left| D_2(p)\tilde{G}D_1(p)\left(\tilde{G} - \tilde{G}_n\right) \right|}{\left| D_2(p)\tilde{G} \cdot D_2(p)\tilde{G}_n \right|} \\
&\le \frac{2\|D_1(p)\|_1 \cdot \|D_2(p)\|_1 \cdot \|\tilde{G}\|_\infty \cdot \|\tilde{G} - \tilde{G}_n\|_\infty}{\left| D_2(p)\tilde{G} \cdot D_2(p)\tilde{G}_n \right|} \\
&= \frac{2O(\epsilon_n h_n^{-2})}{\left| D_2(p)\tilde{G} \cdot D_2(p)\tilde{G}_n \right|}
\end{aligned}
\tag{5.12}
$$

as $\|D_i(p)\|_1 \le O(h^{-1})$ and $\tilde{G}$ is bounded. Also, by a similar analysis, we have

$$
\begin{aligned}
|D_2(p)\tilde{G} - g_z(p)| &= O(h_n^2), \\
|D_2(p)\tilde{G} - D_2(p)\tilde{G}_n| &= O(\epsilon_n h_n^{-1}).
\end{aligned}
\tag{5.13}
$$

Therefore, when n is large enough,

$$|\mu(p) - \tilde{\boldsymbol{\mu}}_n(p)| \le O(h_n^2) + |g_z(p)|^{-2}O(\epsilon_n h_n^{-2}) = O(h_n^2 + \epsilon_n h_n^{-2}). \tag{5.14}$$

This implies that $\tilde{\boldsymbol{\mu}}_n(p)$ converges to $\mu(p)$ for each $p \in P$.

Then, since $|\tilde{\boldsymbol{\mu}}_n| = k_n$ is a constant for each n, it follows that $|\mu| = k$ on P and $k = \lim_n k_n$. Also, since $\lim_n h_n = 0$, P is dense in R_1. By the continuity of μ, $|\mu(x)| = k$ for any $x \in R_1$.

It remains to prove that $\mu = k\bar{\varphi}/|\varphi|$ where φ is holomorphic. Assume $k \ne 0$. Now, we fix a point cloud P_n and consider the matrix L which is the approximation of the Laplace-Beltrami operator. From the error analysis of the MLS method, for

any interior point $p = p_i \in P_n$,

$$
\begin{aligned}
&\left| (\Delta\mu(p)) \, \overline{\mu(p)} - (L_i \tilde{\boldsymbol{\mu}}_n) \, \overline{\tilde{\boldsymbol{\mu}}_n(p)} \right| \\
&\leq \|\Delta\mu\|_\infty \cdot |\mu(p) - \tilde{\boldsymbol{\mu}}_n(p)| + k_n |\Delta\mu(p) - L_i \tilde{\boldsymbol{\mu}}_n| \\
&= \|\Delta\mu\|_\infty O(h_n^2 + \epsilon_n h_n^{-2}) + k_n O(h_n) \\
&= O(h_n^2 + \epsilon_n h_n^{-2}).
\end{aligned}
\tag{5.15}
$$

On the other hand, by the definition of PCT-map, for any n,

$$
(L_i \tilde{\boldsymbol{\mu}}_n) \, \overline{\tilde{\boldsymbol{\mu}}_n(p)} \in \mathbb{R}.
\tag{5.16}
$$

Thus, $(\Delta\mu(p)) \, \overline{\mu(p)} \in \mathbb{R}$ if p is an interior point of P_n.

Consider an arbitrary point $p \in R_1$. If $p \in P - \partial R_1$, then the above conclusion holds. If $p \in P \cap \partial R_1$ or $p \in R_1 - P$, since P is dense in R_1, by the continuity of μ and the Laplace-Beltrami operator, the above conclusion also holds. Therefore, $(\Delta\mu(p)) \, \overline{\mu(p)} \in \mathbb{R}$ for any $p \in R_1$.

Let $\mu = k(\rho + i\tau)$. By direct calculation,

$$
0 = \frac{1}{k^2} Im\left((\Delta\mu) \, \overline{\mu}\right) = \rho\Delta\tau - \tau\Delta\rho.
\tag{5.17}
$$

By taking $u = \rho$, and $v = -\tau$ in Lemma 5.1, there exists a real-valued function f such that $\varphi = f \cdot (\rho - i\tau)$ is holomorphic. Moreover, $\mu = k\bar{\varphi}/|\varphi|$. It follows that g is a Teichmüller mapping. $\square$

From the above theorem, We have established the notion of the planar PCT-maps on planar disk-type point clouds, and proved the consistency between planar PCT-maps and the continuous T-maps. Then, the next step is to define the Teichmüller mappings between general disk-type point cloud surfaces. This generalization can be easily achieved using a similar idea for generalizing planar quasi-conformal point cloud mappings in [35], to which we refer the readers for more details.

The generalization is outlined as follows. Let $\mathbf{f}$ be a point cloud mapping between two point cloud surfaces P_1 and P_2, and $\boldsymbol{\phi}_i$ be the ϵ-conformal parameterization of P_i defined in [35]. Then we define $\mathbf{f}$ to be a *point cloud Teichmüller mapping (PCT-map)* between the two point cloud surfaces if $\boldsymbol{\phi}_2^{-1} \circ \mathbf{f} \circ \boldsymbol{\phi}_1$ is a planar PCT-map.

Using the above definition, a property analogous to Proposition 5.2 can be easily derived for the PCT-maps between two general point cloud surfaces with underlying disk-type Riemann surfaces. We omit the details here.

In conclusion, in order to compute the PCT-maps between two point cloud surfaces, we can simplify the problem by first computing the point cloud conformal parameterizations onto planar domains and then the planar PCT-maps. In the next section, we develop efficient and accurate algorithms for computing the PCT-maps between feature-endowed point clouds.

6. Our proposed TEMPO method for computing PCT-maps. In this section, we describe our proposed TEMPO method for computing Teichmüller mappings between point clouds with disk topology under prescribed feature correspondences. Our proposed TEMPO method primarily consists of four parts:

1. A novel hybrid quasi-conformal reconstruction scheme for computing quasi-conformal mappings on point clouds.

2. An improved approximation of the differential operators arisen from the computation of conformal and quasi-conformal mappings on point clouds with disk topology.
3. Teichmüller parameterization of disk-type point clouds with landmark constraints.
4. Shape analysis of disk-type point clouds via Teichmüller metric.

In the following, we describe the aforementioned parts in details.

6.1. A hybrid quasi-conformal reconstruction scheme for computing quasi-conformal mappings on planar point clouds. Quasi-conformal mappings have been extensively studied on general smooth surfaces and meshes but not on point clouds before. In this subsection, we develop a novel hybrid scheme for computing quasi-conformal mappings on point clouds with a prescribed point cloud Beltrami coefficient (PCBC).

In the continuous case, there are several existing methods to compute quasi-conformal mappings associated with a given feasible Beltrami coefficient $\mu = \rho + i\tau$. One method is to directly solve the Beltrami equation (4.2). Note that the Beltrami equation (4.2) is equivalent to the following linear system

$$\begin{cases} v_y = \alpha_1 u_x + \alpha_2 u_y; \\ -v_x = \alpha_2 u_x + \alpha_3 u_y, \end{cases} \tag{6.1}$$

where

$$\begin{aligned} \alpha_1 &= \frac{(1-\rho)^2 + \tau^2}{1 - \rho^2 - \tau^2}; \\ \alpha_2 &= \frac{-2\tau}{1 - \rho^2 - \tau^2}; \\ \alpha_3 &= \frac{(1+\rho)^2 + \tau^2}{1 - \rho^2 - \tau^2}. \end{aligned} \tag{6.2}$$

Another approach for computing quasi-conformal mappings is the *Linear Beltrami Solver (LBS)* proposed by Lui et al. in [31]. The solver was applied for computing quasi-conformal mappings between planar triangular meshes. The major step of LBS is to solve the following elliptic PDEs

$$\nabla \cdot \left(A \begin{pmatrix} u_x \\ u_y \end{pmatrix} \right) = 0 \text{ and } \nabla \cdot \left(A \begin{pmatrix} v_x \\ v_y \end{pmatrix} \right) = 0 \tag{6.3}$$

where $A = \begin{pmatrix} \alpha_1 & \alpha_2 \\ \alpha_2 & \alpha_3 \end{pmatrix}$ is a symmetric positive definite matrix with $\det(A) = 1$. These two PDEs can be easily derived from Equation (6.1) by taking the divergence on both sides.

The abovementioned methods work well on triangular meshes. However, for the case of point clouds, solely solving the system (6.1) may not result in any meaningful solutions. More specifically, the solution to the system (6.1) is highly sensitive to the input PCBC. If the PCBC is not sufficiently close to a feasible Beltrami coefficient in the continuous case, the system (6.1) may not lead to a quasi-conformal diffeomorphism for point clouds. Also, the solution to the system (6.3) may be associated with a PCBC which is largely different from the input Beltrami coefficients. In other words, the accuracy of the system (6.3) hinders the computation of quasi-conformal mappings

on point clouds. Therefore, we cannot directly apply any of the above methods to compute quasi-conformal mappings on point clouds. To alleviate the abovementioned issues, we propose a hybrid quasi-conformal reconstruction scheme for point clouds by a fusion of the two methods.

Instead of merely adapting either of the two approaches, we consider taking a balance between them in order to achieve a practical scheme for computing quasi-conformal mappings on point clouds. Specifically, we consider solving the following hybrid PDE for computing quasi-conformal mappings:

$$
\begin{pmatrix} \alpha_1 \partial_x + \alpha_2 \partial_y & -\partial_y \\ \alpha_2 \partial_x + \alpha_3 \partial_y & \partial_x \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + \gamma \begin{pmatrix} \mathcal{L} & O \\ O & \mathcal{L} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = 0, \qquad (6.4)
$$

where $\mathcal{L}$ is the generalized Laplace-Beltrami operator, and γ is a user-defined positive parameter.

To solve the hybrid PDE (6.4) on point clouds, we first discretize the system (6.1) and denote it by

$$
M_1(\boldsymbol{\mu}) \begin{pmatrix} U \\ V \end{pmatrix} = 0. \qquad (6.5)
$$

Similarly, the system (6.3) is discretized and denoted by

$$
M_2(\boldsymbol{\mu}) \begin{pmatrix} U \\ V \end{pmatrix} = 0, \qquad (6.6)
$$

where

$$
M_2(\boldsymbol{\mu}) = \begin{pmatrix} M_3(\boldsymbol{\mu}) & O \\ O & M_3(\boldsymbol{\mu}) \end{pmatrix}, \qquad (6.7)
$$

and $M_3(\boldsymbol{\mu})$ is the approximation of the generalized Laplace-Beltrami operator. Here the discretization methods are to be determined. The details of the abovementioned discretizations are described in Section 6.2.1 and Section 6.2.2.

Finally, to compute a quasi-conformal mapping with a given PCBC $\boldsymbol{\mu}$, we propose to solve the hybrid system

$$
(M_1(\boldsymbol{\mu}) + \gamma M_2(\boldsymbol{\mu})) \begin{pmatrix} U \\ V \end{pmatrix} = 0. \qquad (6.8)
$$

One advantage of our proposed hybrid scheme is that it is theoretically supported. Since both of the system (6.1) and the system (6.3) aim to compute quasi-conformal mapping f with the prescribed Beltrami coefficient μ, this mapping $f = (u, v)^T$ is theoretically guaranteed to be a solution to the hybrid PDE (6.4).

Moreover, our proposed scheme overcomes the drawbacks of each of the two approaches with the aid of the other approach. On one hand, the system (6.5) produces accurate results but it is highly unstable. By including the second term M_2 in Equation (6.8), we significantly stabilize the computations. On the other hand, the system (6.6) is stable but not accurate. The first term M_1 in Equation (6.8) helps boosting up the accuracy of the computation. In summary, with a fusion of the two approaches, both the accuracy and the stability of the computation of quasi-conformal mappings on point clouds can be improved.

It is noteworthy that the discretization schemes for the system (6.1) and the system (6.3) on point clouds are crucial to the hybrid equation (6.8). In the following subsection, two discretization schemes are proposed to accurately compute M_1 and M_2 on point clouds.

6.2. Approximating the differential operators in Equation (6.1) and (6.3). In this part, we present the numerical schemes to respectively approximate the first order PDE system (6.1) and the generalized Laplace equation (6.3) on disk-type point clouds.

For the discretization of Equation (6.1), we adopt a numerical scheme which involves the moving least square (MLS) method [27, 28, 5]. For the discretization of the generalized Laplace equation (6.3), we propose a combined scheme which involves both the MLS method [27, 28, 5] and the local mesh method [24]. It is noteworthy that the choice of the weight function w in the MLS method is crucial for the accuracy of the approximations. In this work, we adopt the Gaussian weight function proposed by Choi et al. [5]:

$$w(d) = \begin{cases} 1 & \text{if } d = 0 \\ \frac{1}{K}\exp(\frac{-\sqrt{K}d^2}{D^2}) & \text{if } d \neq 0 \end{cases}, \qquad (6.9)$$

where K is the number of points in the chosen neighborhood, and D is the maximal distance of the neighborhood.

The details of the discretization schemes are discussed below.

6.2.1. Approximating the Beltrami equation on planar point clouds. We first present a scheme to discretize the Beltrami equation on point clouds for computing quasi-conformal mappings. Instead of approximating the ordinary Beltrami Equation (4.2), we aim to approximate the equivalent system (6.1) on a point cloud P.

For each point $p \in P$, the MLS method constructs an approximating function f_p near p. To satisfy the system (6.1) on the point p, a straightforward calculation yields the following equation

$$\begin{pmatrix} q_2^T(p)A_p \\ -q_1^T(p)A_p \end{pmatrix} V = A(p) \begin{pmatrix} q_1^T(p)A_p \\ q_2^T(p)A_p \end{pmatrix} U. \qquad (6.10)$$

The above system can be reformulated as a linear system of the form $M_1(\boldsymbol{\mu})[U^T, V^T]^T = 0$, where $M_1(\boldsymbol{\mu})$ is a $2N \times 2N$ matrix, with N being the number of points in P. More explicitly, for any $i = 1, 2, \cdots, N$, we have

$$\begin{pmatrix} (M_1(\boldsymbol{\mu}))_i \\ (M_1(\boldsymbol{\mu}))_{i+N} \end{pmatrix} = \begin{pmatrix} \alpha_1(p_i)q_1^T(p_i) + \alpha_2(p_i)q_2^T(p_i) & -q_2^T(p_i) \\ \alpha_2(p_i)q_1^T(p_i) + \alpha_3(p_i)q_2^T(p_i) & q_1^T(p_i) \end{pmatrix} \begin{pmatrix} A_{p_i} & O \\ O & A_{p_i} \end{pmatrix},$$
$$(6.11)$$

where E_i denotes the i-th row of any matrix E. The following lemma establishes a theoretical guarantee for our proposed scheme.

LEMMA 6.1. *Let P be a quasi-uniform point cloud sampled from a domain that satisfies an interior cone condition. Suppose $\mathbf{f} = (\mathbf{u}, \mathbf{v})^T$ is a point cloud mapping defined on P with well-defined discrete diffuse PCBC. Let $\boldsymbol{\sigma}$ be a complex-valued point cloud function. Then $\boldsymbol{\sigma}$ is the discrete diffuse PCBC of $\mathbf{f}$ if and only if $M_1(\boldsymbol{\sigma})[U^T, V^T]^T = 0$.*

Proof. Note that $\mathbf{f}$ satisfies $M_1(\boldsymbol{\sigma})[U^T, V^T]^T = 0$ if and only if the Equation (6.10) holds for all $p \in P$, which is true if and only if

$$(q_1(p) + iq_2(p))^T A_p(U + iV) = \boldsymbol{\sigma}(p)(q_1(p) - iq_2(p))^T A_p(U + iV) \qquad (6.12)$$

holds for all $p \in P$. Since the discrete diffuse PCBC is well-defined for all points in P, by direct calculation, the above equation holds if and only if $\boldsymbol{\sigma}(p)$ is the diffuse PCBC of $\mathbf{f}$ on p. This completes the proof. □

By the above lemma, the solution to the linear system $M_1(\boldsymbol{\mu})[U^T, V^T]^T = 0$ is associated with a diffuse PCBC which is consistent with the input Beltrami coefficient $\boldsymbol{\mu}$, if it is admissible. This ensures the accuracy of the computation of quasi-conformal mappings on point clouds.

6.2.2. Improving the approximation of the generalized Laplacian for planar point clouds with disk topology. Next, we introduce our proposed approximation scheme for the generalized Laplace equation (6.3) on point clouds with disk topology. Note that the generalized Laplacian is closely related to the Laplace-Beltrami operator. For approximating the Laplace-Beltrami operator on point clouds, a number of works have been established with the aid of the moving least square (MLS) method [27, 28, 5]. Therefore, it is natural to consider the MLS method for the generalized Laplacian. In the following, we propose an approximation scheme for the generalized Laplacian on disk-type planar point clouds by combining the MLS approach in [5] and the local mesh approach in [24].

The existing MLS approaches work well for interior points in disk-type domains. However, unlike the approximations of the Laplace-Beltrami operator on point clouds and the generalized Laplace-Beltrami operator on meshes, the Beltrami coefficients for general point cloud mappings are not locally constant. Therefore, the matrix A in Equation (6.3) is not constant. Consequently, we need to take its derivatives into considerations. Specifically, we have

$$
\begin{aligned}
&\nabla \cdot \left(A \begin{pmatrix} u_x \\ u_y \end{pmatrix} \right) \\
&= \partial_x(\alpha_1 u_x + \alpha_2 u_y) + \partial_y(\alpha_2 u_x + \alpha_3 u_y) \\
&= (\partial_x \alpha_1 + \partial_y \alpha_2) u_x + (\partial_x \alpha_2 + \partial_y \alpha_3) u_y + \alpha_1 u_{xx} + 2\alpha_2 u_{xy} + \alpha_3 u_{yy}.
\end{aligned}
\tag{6.13}
$$

By applying the MLS method on the derivatives of α_i and u, the approximation linear system can be obtained. After certain calculations, it follows that if the point $p = p_i$ is an interior point, then the i-th row of matrix $M_3(\boldsymbol{\mu})$ is given by

$$
\begin{aligned}
(M_3(\boldsymbol{\mu}))_i = {} & [\alpha_1(P)]^T B_{11}(p) + [\alpha_2(P)]^T (B_{12}(p) + B_{21}(p)) + [\alpha_3(P)]^T B_{22}(p) \\
& + 2\alpha_1(p)(A_p)_4 + 2\alpha_2(p)(A_p)_5 + 2\alpha_3(p)(A_p)_6
\end{aligned}
$$

where

$$
B_{jl}(p) = A_p^T q_j(p) q_l^T(p) A_p,
$$

$[\alpha_j(P)]$ denotes a vector whose l-th element is $\alpha_j(p_l)$, and E_j denotes the j-th row of matrix E.

However, for the boundary points $p \in \partial\Omega$, the abovementioned approach cannot accurately approximate the operator, since the second order MLS method requires more points to provide an accurate result. To alleviate the problem, we apply the local mesh method [24]. For each boundary point $p_i \in \partial\Omega$, a 1-ring structure $\mathcal{T}_i$ is constructed using the Delaunay triangulation. If the point p_j is not included in the 1-ring, the entry $(M_3(\boldsymbol{\mu}))_{ij}$ is zero. Otherwise, the entry is given by

$$
(M_3(\boldsymbol{\mu}))_{ij} =
\begin{cases}
\displaystyle\sum_{T=\{i,j,k\}\in\mathcal{T}_i} \frac{1}{|T|}(p_k - p_i)^T A(T)(p_k - p_j) & \text{if } j \neq i \\[2ex]
\displaystyle\sum_{T=\{i,l,k\}\in\mathcal{T}_i} \frac{1}{|T|}(p_k - p_l)^T A(T)(p_k - p_l) & \text{if } j = i
\end{cases}
\tag{6.14}
$$

where

$$A(\{i, j, k\}) = \frac{1}{3}(A(p_i) + A(p_j) + A(p_k)). \tag{6.15}$$

Our proposed combined method is shown to produce more accurate approximations than the conventional approaches for the generalized Laplace-Beltrami operator on disk-type point clouds. Various numerical experiments are provided in Section 7 to demonstrate the effectiveness of our proposed approximation scheme.

6.3. Teichmüller parameterization of disk-type point clouds with landmark constraints. Let $\mathcal{M}$ be a simply-connected open surface in $\mathbb{R}^3$, and P be a point cloud sampled from $\mathcal{M}$. We aim to compute a parameterization $\varphi : P \to R \subset \mathbb{R}^2$ with suitable boundary conditions, such that φ approximates a Teichmüller extremal mapping (T-map) from $\mathcal{M}$ to R. In this paper, we set the target domain R to be a rectangle.

Our proposed method primarily consists of two steps, namely,
1. Conformally parameterizing a disk-type point cloud onto a rectangular planar domain, and
2. Computing a Teichmüller extremal mapping on the rectangular domain with the prescribed landmark constraints.

6.3.1. Conformal parameterization to a rectangle. We first describe the procedure of conformally parameterizing a disk-type point cloud P onto a rectangle $\tilde{R}$.

We start by introducing our proposed conformal parameterization scheme under the continuous setting. In the continuous case, we consider the conformal parameterization of a Riemann surface $\mathcal{M}$. One common surface parameterization method is to compute the harmonic mapping by solving the Laplace equation with certain boundary conditions. As an initialization, the first step of our proposed method is to compute the harmonic mapping $\phi_0 : \mathcal{M} \to D$ that satisfies

$$\Delta\phi_0 = 0 \tag{6.16}$$

with arc-length parameterization boundary constraints. Note that the mapping ϕ_0 may not be conformal due to the arc-length parameterization boundary conditions. The conformality distortions are corrected in the following two steps.

Before introducing the two steps, we explain the motivation for developing them. After obtaining the initial map $\phi_0 : \mathcal{M} \to D$, we aim to find a mapping from the unit disk D to a rectangle $\tilde{R}$. By scaling, we assume that $\tilde{R}$ is with height h and width 1. It is noteworthy that in general, $h \neq 1$. Our ultimate goal is to determine a suitable h such that there exists a conformal mapping $\phi : \mathcal{M} \to \tilde{R}$ with four specified boundary points mapped to the four corners of $\tilde{R}$. The existence of such h and ϕ is guaranteed by the Riemann mapping theorem. By the Riemann mapping theorem, a conformal mapping from a simply-connected open bounded surface $\mathcal{M}$ to the unit disk D exists. Moreover, the mapping is unique if three points are fixed on the boundary. Analogously, by restricting four given points to be mapped to four corner points, there exists a positive number h such that the unique conformal mapping exists from $\mathcal{M}$ to the rectangle $\tilde{R}$ with height h and unit width. To explicitly compute h and the desired conformal mapping ϕ, our strategy is to find a quasi-conformal mapping from the unit disk D to the unit square R, and then develop a method to adjust the height of R to an optimal h.

In the second step of our proposed method, we compute the desired quasi-conformal mapping from unit disk D to the unit square R. Recall that we have obtained the initial map $\phi_0 : \mathcal{M} \to D$. By quasi-conformal theory, if two quasi-conformal mappings f and g are associated with the same Beltrami coefficients, then $f \circ g^{-1}$ is conformal. Therefore, the problem boils down to solving for a quasi-conformal mapping from D to $\tilde{R}$ with Beltrami coefficient $\mu(\phi_0^{-1})$. The following proposition guarantees that such a quasi-conformal mapping can be obtained by vertically scaling the mapping $\phi_1 = (u_1, v_1)^T : D \to R$, which solves the generalized Laplace equation (6.3).

Proposition 6.1. *Let $f = u + iv$ be a quasi-conformal mapping from the unit disk D to a rectangle $\tilde{R}$ with height h and unit width. Denote the Beltrami coefficient associated with f by μ. Then, u and v/h satisfy the generalized Laplace equation (6.3) with the Beltrami coefficient μ.*

Proof. Let g be the metric tensor on the unit disk D induced by μ. Then $f_1 : (D, g) \to R$ is a conformal map, where $f_1(x) = f(x)$. Therefore, u, v are solutions to the Laplace equation defined on (D, g). More specifically,

$$\frac{1}{\det(g)} \partial_i (\sqrt{\det(g)} g^{ij} \partial_j u) = 0, \tag{6.17}$$

$$\frac{1}{\det(g)} \partial_i (\sqrt{\det(g)} g^{ij} \partial_j v) = 0. \tag{6.18}$$

Moreover, since the Laplace-Beltrami operator is linear, u and v/h are also solutions to the Laplace equation with respect to the metric tensor. This implies that u and v/h are solutions to the PDE

$$\nabla \cdot (\sqrt{\det(g)}(g^{ij})\nabla x) = 0. \tag{6.19}$$

On the other hand, by direct calculation, the coefficient matrix A in the generalized Laplace-Beltrami operator with Beltrami coefficient μ satisfies

$$A = \sqrt{\det(g)}(g^{ij}). \tag{6.20}$$

More explicitly, we have

$$A = \frac{1}{1 - |\mu|^2} \begin{pmatrix} 1 + |\mu|^2 - 2Re(\mu) & -2Im(\mu) \\ -2Im(\mu) & 1 + |\mu|^2 + 2Re(\mu) \end{pmatrix}. \tag{6.21}$$

Therefore, u and v/h are solutions of the generalized Laplace equation with Beltrami coefficient μ from the unit disk D to the unit square with specific boundary conditions. □

With the guarantee by the above proposition, the final result $\phi = (u_1, hv_1)^T \circ \phi_0$ can be obtained by finding a quasi-conformal mapping $\phi_1 = (u_1, v_1)^T$ from the unit disk D to the unit square R, and scaling its height by a scalar h.

In the final step, it remains to determine the scalar h such that $\phi_2 = (u_1, hv_1)^T$ is a quasi-conformal mapping with Beltrami coefficient equals $\mu(\phi_0^{-1})$. To compute such an optimal h, it is natural to consider the following energy minimization problem

$$h = argmin \int_{\mathbb{D}} |\mu(\phi_2) - \mu(\phi_0^{-1})|^2. \tag{6.22}$$

The energy is zero when h is optimal.

After obtaining the optimal h, the desired conformal parameterization $\phi : \mathcal{M} \to \tilde{R}$ is given by

$$\phi = \phi_2 \circ \phi_0. \tag{6.23}$$

To implement the abovementioned procedures on point cloud data, we need to discretize the operators in every step.

In the initialization step, the MLS method with Gaussian weight [5] is applied for the approximation of the Laplace-Beltrami operator Δ on point clouds. Theoretically, for a mapping φ from a planar domain D to the surface $\mathcal{M}$, the Laplace-Beltrami operator of any function $f : \mathcal{M} \to \mathbb{R}$ is a scaling of the generalized Laplace-Beltrami operator with Beltrami coefficient $\mu(\varphi)$ acting on the composition mapping $f \circ \varphi$ up to some scaling factor λ independent of f. In other words,

$$\lambda(\varphi)(\Delta_{\mathcal{M}} f) \circ \varphi = \nabla \cdot (A(\mu(\varphi))\nabla(f \circ \varphi)). \tag{6.24}$$

Instead of solving the Laplace equation, we solve the generalized Laplace system for some suitable mapping φ.

For every point $p \in P$, consider the projection mapping $\mathbf{g}$ to the best fitting plane computed using the PCA algorithm near p. By applying the MLS method to the point cloud mapping $\mathbf{g}^{-1}$, the polynomial function g_1 is obtained such that the image of the mapping $g_2(x,y) = (x, y, g_1(x,y))^T$ is a local approximation of the surface $\mathcal{M}$ near p up to some rigid motion L. Then, by applying the above idea to the mapping $L \circ g_2$, the Laplace equation at the point p is approximated by

$$0 = \nabla \cdot (A(\mu(L \circ g_2))\nabla f_1)|_p = \nabla \cdot (A(\mu(g_2))\nabla f_1)|_p, \tag{6.25}$$

where f_1 is a function approximating the point cloud function $f|_P \circ \mathbf{g}^{-1}$ using the MLS method. The above term can be explicitly expressed in a closed form, from which a linear system can be formulated with the variables $f|_P$. By solving this linear system with arc-length boundary condition, the initial map ϕ_0 is obtained.

In the second step, to approximate the Beltrami coefficient $\mu(\phi_0^{-1})$, the diffuse PCBC defined in [35] is adapted. With a given PCBC, the desired quasi-conformal mapping $\phi_1 = (\mathbf{u}_1, \mathbf{v}_1)^T$ from point cloud P to the unit square R is solved by the generalized Laplace equation (6.6).

In the final step, we aim to simplify the computation of the optimal height h of the rectangle. Recall that the point clouds are assumed to be quasi-uniform. Therefore, we discretize the energy in Equation (6.22) by taking average over all points. Now, it suffices to minimize

$$\|\tilde{\boldsymbol{\sigma}} - \tilde{\boldsymbol{\mu}}\|^2, \tag{6.26}$$

where $\tilde{\boldsymbol{\sigma}}$ and $\tilde{\boldsymbol{\mu}}$ are respectively the discrete diffuse PCBCs of $\phi_2 = (\mathbf{u}_1, h\mathbf{v}_1)^T$ and ϕ_0^{-1}. This discretization largely simplifies the computation of the optimal h.

Our proposed algorithm for computing the conformal parameterizations of disk-type point clouds is summarized in Algorithm 1.

Algorithm 1: Conformal parameterization of disk-type point clouds

---

    Input: A point cloud surface P with disk topology, four boundary points
           $p_i, i = 1, 2, 3, 4$
    Output: The point cloud conformal parameterization ϕ of P to a rectangle $\tilde{R}$
           with all p_i mapped to the four corners

1 Solve the Laplace equation with arc-length parameterization boundary
    constraints and obtain $\phi_0 : P \to D$;

2 Compute the discrete diffuse PCBC $\tilde{\mu}$ of the point cloud mapping ϕ_0^{-1};

3 Solve the generalized Laplace equation (6.6) with the associated PCBC $\tilde{\mu}$, and
    obtain a mapping $\phi_1 = (\mathbf{u}_1, \mathbf{v}_1)^T$ from D to the unit square R, with four
    points p_1, p_2, p_3, p_4 mapped to the corresponding four corners;

4 Adjust the height of R to be h by minimizing $\|\tilde{\sigma} - \tilde{\mu}\|^2$, where $\tilde{\sigma}$ is the
    discrete diffuse PCBC of $(\mathbf{u}_1, h\mathbf{v}_1)^T$;

5 The conformal parameterization is given by $\phi = (\mathbf{u}_1, h\mathbf{v}_1)^T \circ \phi_0$ from P to a
    rectangle $\tilde{R}$ with height h and width 1.

6.3.2. Teichmüller parameterization of disk-like point clouds with landmark constraints.

We now aim to develop a method for computing landmark-matching Teichmüller parameterization of point clouds with disk topology. By using Algorithm 1 proposed in the last section, every disk-type point cloud can be conformally mapped to a rectangular domain in $\mathbb{R}^2$. Therefore, in this part, we focus on the Teichmüller extremal mapping of planar point clouds with prescribed landmark constraints. In [32], Lui et al. proposed a method for computing T-maps on triangular meshes. The convergence of the method under the continuous setting was proved in [33]. In this work, we develop an iterative algorithm for PCT-maps on point clouds and study the convergence properties.

Recall that searching for a quasi-conformal mapping is equivalent to searching for a specific Beltrami coefficient in the space of complex-valued functions. In this section, we propose an iterative algorithm to solve for a T-map on point clouds by manipulating the Beltrami coefficients. Specifically, the desired Beltrami coefficients should be with constant norms and satisfy the specific form for the argument part. We start our algorithm by initializing the mapping to be the identity map. Then, in each iteration, four steps as introduced below are performed on the associated discrete diffuse PCBC of the mapping.

In the first step of each iteration, we compute the Beltrami coefficient $\tilde{\mu}_n$ of a given point cloud mapping $\mathbf{f}_n$ using the definition of PCBC in [35].

In the second step, we aim to manipulate the norm of the PCBC so that the new PCBC gets closer to the Beltrami coefficient of a T-map. Specifically, a suitable constant k_n is chosen to be the norm of the new PCBC. Here, we consider taking the average value of the norms of $\tilde{\mu}_n$ at the points where the norms are less than 1. In other words, we take

$$k_n = \frac{\sum_{p \in S_n} |\tilde{\mu}_n(p)|}{|S_n|} \tag{6.27}$$

where

$$S_n = \{p \in P : |\tilde{\mu}_n(p)| < 1\}. \tag{6.28}$$

This step guarantees that the new norm is a feasible norm.

In the third step, we update the argument part of the PCBC by using that of $\tilde{\boldsymbol{\mu}}_n$. Define $\boldsymbol{\nu}_n$ by

$$\boldsymbol{\nu}_n(p_i) = \begin{cases} \dfrac{\tilde{\boldsymbol{\mu}}_n(p_i)}{|\tilde{\boldsymbol{\mu}}_n(p_i)|} & \text{if } \tilde{\boldsymbol{\mu}}_n(p_i) \neq 0, \\ 1 & \text{otherwise.} \end{cases} \tag{6.29}$$

Then, we apply the weighted Laplacian smooth operator and the normalization operator on $\boldsymbol{\nu}_n$ and obtain

$$\boldsymbol{\tau}_n(p_i) = \begin{cases} \dfrac{L_i \boldsymbol{\nu}_n}{|L_i \boldsymbol{\nu}_n|} & \text{if } L_i \boldsymbol{\nu}_n \neq 0, \\ \boldsymbol{\nu}_n(p_i) & \text{otherwise.} \end{cases} \tag{6.30}$$

Here L is the approximation of the Laplace-Beltrami operator on point clouds as described in Section 6.2.2. With the new norm k_n and the new argument part $\boldsymbol{\tau}_n$, we construct a new complex-valued point cloud function

$$\boldsymbol{\sigma}_n = k_n \boldsymbol{\tau}_n. \tag{6.31}$$

In the last step, a new mapping $\mathbf{f}_{n+1}$ is computed by solving the hybrid equation (6.8) with the PCBC $\boldsymbol{\sigma}_n$ and the prescribed landmark constraints. It is noteworthy that since $\boldsymbol{\sigma}_n$ may not be a feasible PCBC and the landmark constraints are enforced, the resulting mapping $\mathbf{f}_{n+1}$ may not be exactly associated with $\boldsymbol{\sigma}_n$. In this case, this step can be regarded as a projection of $\boldsymbol{\sigma}_n$ onto the space of all feasible PCBCs.

By repeating the abovementioned steps, the mapping converges to the landmark-aligned Teichmüller parameterization. The entire algorithm is summarized in Algorithm 2.

After describing our proposed algorithm for computing landmark-matching Teichmüller mappings on point clouds, we prove that the limit function obtained by our algorithm is indeed a PCT-map.

Proposition 6.2. *Let P be a quasi-uniform point cloud sampled from a domain which satisfies the interior cone condition. Assume that the sequence $\mathbf{f}_n$ in Algorithm 2 converges to a point cloud mapping $\mathbf{f}$, and its discrete diffuse PCBC $\tilde{\boldsymbol{\mu}}$ exists. Further assume that $\lim_n \gamma_n = 0$. Then $\mathbf{f}$ is a PCT-map. Moreover, both $\tilde{\boldsymbol{\mu}}_n$ and $\boldsymbol{\sigma}_n$ converge to $\tilde{\boldsymbol{\mu}}$.*

Proof. First, we prove that $\tilde{\boldsymbol{\mu}}_n$ converges to $\tilde{\boldsymbol{\mu}}$. Now we adapt the notation D_1, D_2 as defined in the proof of Proposition 5.2. By definition, for any point $x \in P$,

$$
\begin{aligned}
&|\tilde{\boldsymbol{\mu}}_n(x) - \tilde{\boldsymbol{\mu}}(x)| \\
&= \left| \frac{D_1\left[\begin{pmatrix} U_n \\ V_n \end{pmatrix} - \begin{pmatrix} U \\ V \end{pmatrix}\right] + D_1\begin{pmatrix} U \\ V \end{pmatrix}}{D_2\left[\begin{pmatrix} U_n \\ V_n \end{pmatrix} - \begin{pmatrix} U \\ V \end{pmatrix}\right] + D_2\begin{pmatrix} U \\ V \end{pmatrix}} - \frac{D_1\begin{pmatrix} U \\ V \end{pmatrix}}{D_2\begin{pmatrix} U \\ V \end{pmatrix}} \right|.
\end{aligned}
\tag{6.32}
$$

By error analysis in MLS, for $j = 1, 2$,

$$|q_j^T(x) A_x (U_n - U)| \leq \|U_n - U\|_\infty \|q_j^T(x) A_x\|_1 \leq \|U_n - U\|_\infty O(h^{-1}), \tag{6.33}$$

Algorithm 2: Landmark-matching Teichmüller parameterization of planar point clouds

---

    Input: A planar point cloud P, target rectangle R and the landmark constraints

    Output: The landmark-matching Teichmüller parameterization $\mathbf{f} : P \to R$

1 Initialize the mapping $\mathbf{f}_0$ to be identity map;

2 while $\|F_n - F_{n-1}\|_2 \geq \epsilon$ **do**

3   Calculate the discrete diffuse PCBC $\tilde{\boldsymbol{\mu}}_n$ of $\mathbf{f}_n$;

4   Compute $k_n = \frac{\sum_{p \in S_n} |\tilde{\boldsymbol{\mu}}_n(p)|}{|S_n|}$, where $S_n = \{p \in P : |\tilde{\boldsymbol{\mu}}_n(p)| < 1\}$;

5   Calculate the argument part of $\tilde{\boldsymbol{\mu}}_n$ and denote it as $\boldsymbol{\nu}_n$;

6   Construct a complex valued point cloud function $\boldsymbol{\tau}_n$ by

$$
\boldsymbol{\tau}_n(p_i) = \begin{cases} \dfrac{L_i \boldsymbol{\nu}_n}{|L_i \boldsymbol{\nu}_n|} & \text{if } L_i \boldsymbol{\nu}_n \neq 0, \\[2mm] \boldsymbol{\nu}_n(p_i) & \text{otherwise} \end{cases}
$$

     where $L = M_2(0)$ is the approximation of the Laplace-Beltrami operator by our proposed scheme described in Section 6.2.2;

7   Construct a new complex valued point cloud function $\boldsymbol{\sigma}_n = k_n \boldsymbol{\tau}_n$;

8   Compute the point cloud mapping $\mathbf{f}_{n+1}$ by solving the hybrid equation (6.8) with a given parameter $\gamma_n \in [0, \infty)$, the input BC $\boldsymbol{\sigma}_n$ and the prescribed landmark constraints;

9   Denote the matrix form of $\mathbf{f}_{n+1}$ by F_{n+1};

10   Update n by $n + 1$;

11 Obtain the final mapping $\mathbf{f} = \mathbf{f}_n$.

---

where h is the fill distance of P and hence constant. Since $\mathbf{f}_n$ converges to $\mathbf{f}$ and both of them are defined on a finite set P, we have $\|U_n - U\|_\infty \to 0$, which implies that $|q_j^T(x) A_x (U_n - U)| \to 0$. A similar result also holds for V. Therefore, for $j = 1, 2$,

$$
\lim_n \left| D_j \left[\begin{pmatrix} U_n \\ V_n \end{pmatrix} - \begin{pmatrix} U \\ V \end{pmatrix} \right] \right| = 0. \tag{6.34}
$$

This shows that $\tilde{\boldsymbol{\mu}}_n \to \tilde{\boldsymbol{\mu}}$.

Then, we prove that $\boldsymbol{\sigma}_n$ also converges to $\tilde{\boldsymbol{\mu}}$. Recall that in Algorithm 2, $\mathbf{f}_{n+1}$ solves the hybrid equation (6.8) with input PCBC $\boldsymbol{\sigma}_n$. Let

$$
E_n = -\gamma_n M_2(\boldsymbol{\sigma}_n) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix}. \tag{6.35}
$$

Then

$$
M_1(\boldsymbol{\sigma}_n) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} = E_n. \tag{6.36}
$$

By direct calculation, for any $x = p_j \in P$,

$$
\begin{aligned}
&D_1(x) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} \\
=&q_1^T(x) A_x U_{n+1} + i q_2^T(x) A_x U_{n+1} + i q_1^T(x) A_x V_{n+1} - q_2^T(x) A_x V_{n+1} \\
=&q_1^T(x) A_x U_{n+1} + i q_2^T(x) A_x U_{n+1} - i(\alpha_2(\boldsymbol{\sigma}_n|_x) q_1^T(x) + \alpha_3(\boldsymbol{\sigma}_n|_x) q_2^T(x)) A_x U_{n+1} \\
&- (\alpha_1(\boldsymbol{\sigma}_n|_x) q_1^T(x) + \alpha_2(\boldsymbol{\sigma}_n|_x) q_2^T(x)) A_x U_{n+1} + i(E_n)_{j+N} + (E_n)_j \\
=&\left(2\frac{\boldsymbol{\sigma}_n(x) - |\boldsymbol{\sigma}_n(x)|^2}{1 - |\boldsymbol{\sigma}_n(x)|^2} q_1(x) - 2i \frac{\boldsymbol{\sigma}_n(x) + |\boldsymbol{\sigma}_n(x)|^2}{1 - |\boldsymbol{\sigma}_n(x)|^2} q_2(x) \right)^T A_x U_{n+1} \\
&+ i(E_n)_{j+N} + (E_n)_j.
\end{aligned}
$$
(6.37)

Similarly, the following equation also holds.

$$
\begin{aligned}
&D_2(x) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} \\
=&\left(2\frac{1 - \overline{\boldsymbol{\sigma}_n(x)}}{1 - |\boldsymbol{\sigma}_n(x)|^2} q_1(x) - 2i \frac{1 + \overline{\boldsymbol{\sigma}_n(x)}}{1 - |\boldsymbol{\sigma}_n(x)|^2} q_2(x) \right)^T A_x U_{n+1} + i(E_n)_{j+N} - (E_n)_j.
\end{aligned}
$$
(6.38)

Then,

$$
\boldsymbol{\sigma}_n(x) = \frac{D_1(x) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} - i(E_n)_{j+N} - (E_n)_j}{D_2(x) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} - i(E_n)_{j+N} + (E_n)_j}.
$$
(6.39)

Since the point cloud P is fixed and $|\boldsymbol{\sigma}_n(x)| \leq k_n < 1$ is bounded, the matrix $M_2(\boldsymbol{\sigma}_n)$ is bounded. Also, since $\mathbf{f}_n$ is a point cloud mapping within a fixed rectangle, the vectors U_n and V_n are also bounded. Therefore, E_n converges to the zero vector since γ_n converges to zero. Moreover, from Equation (6.34) and the assumption that $\tilde{\boldsymbol{\mu}}$ is well-defined, we conclude that $|D_2(x)[U_{n+1}^T, V_{n+1}^T]^T|$ is bounded from below when n is large, and the lower bound is a positive number. Thus, the right hand side in Equation (6.39) is well-defined, and it converges to $\lim_n \tilde{\boldsymbol{\mu}}_{n+1}(x) = \tilde{\boldsymbol{\mu}}(x)$. This shows that $\boldsymbol{\sigma}_n$ converges to $\tilde{\boldsymbol{\mu}}$.

Since $|\boldsymbol{\sigma}_n(x)| = k_n < 1$ is a constant for any $x \in P$, $k := \lim_n k_n = |\tilde{\boldsymbol{\mu}}(x)|$ is also a constant.

Let p_i be an arbitrary interior point. It remains to prove that $(L_i \tilde{\boldsymbol{\mu}}) \overline{\tilde{\boldsymbol{\mu}}(p_i)} \in \mathbb{R}$. The statement is trivial if $L_i \tilde{\boldsymbol{\mu}} = 0$. Hence, we only consider the case when $L_i \tilde{\boldsymbol{\mu}} \neq 0$ and $k \neq 0$. Let $\tilde{\boldsymbol{\mu}}_n = \tilde{\boldsymbol{\mu}}_{n,1} + i \tilde{\boldsymbol{\mu}}_{n,2}$, $\tilde{\boldsymbol{\mu}} = \tilde{\boldsymbol{\rho}}_1 + i \tilde{\boldsymbol{\rho}}_2$, and $\boldsymbol{\sigma}_n = \boldsymbol{\sigma}_{n,1} + i \boldsymbol{\sigma}_{n,2}$. Without loss of generality, let $\tilde{\boldsymbol{\mu}}_2(p_i) \neq 0$. Then, when n is large enough,

$$
\frac{\tilde{\boldsymbol{\rho}}_1(p_i)}{\tilde{\boldsymbol{\rho}}_2(p_i)} = \lim_n \frac{\boldsymbol{\sigma}_{n,1}(p_i)}{\boldsymbol{\sigma}_{n,2}(p_i)} = \lim_n \frac{L_i(\tilde{\boldsymbol{\mu}}_{n,1}/|\tilde{\boldsymbol{\mu}}_n|)}{L_i(\tilde{\boldsymbol{\mu}}_{n,2}/|\tilde{\boldsymbol{\mu}}_n|)} = \frac{L_i(\tilde{\boldsymbol{\rho}}_1/k)}{L_i(\tilde{\boldsymbol{\rho}}_2/k)} = \frac{L_i \tilde{\boldsymbol{\rho}}_1}{L_i \tilde{\boldsymbol{\rho}}_2}
$$
(6.40)

Therefore, $(L_i \tilde{\boldsymbol{\rho}}_1) \tilde{\boldsymbol{\rho}}_2(p_i) = (L_i \tilde{\boldsymbol{\rho}}_2) \tilde{\boldsymbol{\rho}}_1(p_i)$ and the result follows. □

As shown above, if Algorithm 2 converges, then $\mathbf{f}_n$ converges to the PCT-map we desired, and $\tilde{\boldsymbol{\mu}}_n$ and $\boldsymbol{\sigma}_n$ converge to its discrete diffuse PCBC. Moreover, the limit PCBC has a constant norm. This satisfies the requirement about the norm of the Beltrami coefficients of PCT-maps in Definition 5.1. Furthermore, the other requirement $(L_i\tilde{\boldsymbol{\mu}})\overline{\tilde{\boldsymbol{\mu}}(p_i)} \in \mathbb{R}$ in Definition 5.1 is also fulfilled. As a remark, since Proposition 5.2 is valid when the fill distance of a sequence of point clouds converges to zero, it follows that our algorithm produces a more accurate result as the density of the point cloud increases.

In conclusion, we have proposed an effective algorithm for computing landmark-matching PCT-maps on disk-type point clouds with mathematical guarantees. Experimental results of the computation of PCT-maps are presented in Section 7.

6.4. Shape analysis of disk-like point clouds via Teichmüller metric. With our proposed Teichmüller parameterization algorithm, we can easily compute mappings between feature-endowed point clouds. Furthermore, using the Teichmüller metric induced by Teichmüller mappings, we can effectively classify different point clouds. In this section, we present our proposed framework for registering point clouds with landmark constraints and classifying different point clouds.

First, we describe our proposed registration algorithm in the continuous setting. Suppose Ω_1 and Ω_2 are two disk-type Riemann surfaces, and the landmark correspondences are given by

$$p_i \leftrightarrow q_i \tag{6.41}$$

for $i = 1, 2, \cdots, n$. We aim to compute the Teichmüller mapping $f : \Omega_1 \to \Omega_2$ such that $f(p_i) = q_i$ for all $i = 1, 2, \cdots, n$.

We start by computing the conformal parameterizations $g_1 : \Omega_1 \to \tilde{R}_1$, $g_2 : \Omega_2 \to \tilde{R}_2$. Then, we compute the Teichmüller parameterization $h : \tilde{R}_1 \to \tilde{R}_2$ between the two planar domains, with the following landmark correspondences

$$h(g_1(p_i)) = g_2(q_i) \tag{6.42}$$

for $i = 1, 2, \cdots, n$. Finally, a mapping $f : \Omega_1 \to \Omega_2$ can be obtained by a composition of the abovementioned mappings. More specifically, we define

$$f = g_2^{-1} \circ h \circ g_1. \tag{6.43}$$

Apparently, f satisfies

$$f(p_i) = q_i. \tag{6.44}$$

Moreover, f is a Teichmüller mapping. In other words, the Beltrami coefficient associated with f is with constant norm, and its argument part is the conjugate of the argument part of a nonzero holomorphic mapping. This phenomenon can be explained by the composition property of quasi-conformal mappings. Note that since h is a Teichmüller mapping, the Beltrami coefficient μ_h is with constant norm. Also, since g_1, g_2 are conformal mappings, the Beltrami coefficient μ_f, μ_g is zero. By Theorem 4.1,

$$\mu_f = \mu_{g_2^{-1} \circ h \circ g_1} = (\mu_h \circ g_1)\frac{\overline{(\partial_z g_1)}}{\partial_z g_1} = k\frac{\overline{(\partial_z g_1)^2(\varphi \circ g_1)}}{|(\partial_z g_1)^2(\varphi \circ g_1)|} \tag{6.45}$$

where $k = |\mu_h|$ is a constant, and φ is holomorphic. Moreover, since g_1 is conformal, $(\partial_z g_1)^2 (\varphi \circ g_1)$ is holomorphic. Therefore, the registration mapping f is a landmark-matching Teichmüller mapping.

In the discrete case, the computations of the conformal and the Teichmüller parameterizations of point clouds can be respectively achieved by Algorithm 1 and Algorithm 2. The entire registration algorithm is summarized in Algorithm 3.

Algorithm 3: Landmark-matching Teichmüller registration between two point clouds with disk topology

---

Input: Two disk-type point clouds P_1 and P_2, with landmark correspondences $p_i \leftrightarrow q_i$, $p_i \in P_1$, $q_i \in P_2$, $i = 1, 2, ..., n$
Output: The Teichmüller mapping $\mathbf{f} : P_1 \to P_2$ with $\mathbf{f}(p_i) = q_i$ for all i
1 Using Algorithm 1, compute the conformal parameterizations $\mathbf{g}_t$ from P_t to a rectangular domain $\tilde{R}_t$, where $t = 1, 2$;
2 Using Algorithm 2, compute the Teichmüller parameterization $\mathbf{h}$ from $\tilde{R}_1$ to the domain $\tilde{R}_2$, with the landmark constraints $\mathbf{h}(\mathbf{g}_1(p_i)) = \mathbf{g}_2(q_i)$ for all i;
3 Obtain the registration mapping $\mathbf{f} = \mathbf{g}_2^{-1} \circ \mathbf{h} \circ \mathbf{g}_1$.

Furthermore, with the aid of our landmark-constrained Teichmüller registration scheme, a metric shape space can be built for point cloud classifications and shape analyses. More explicitly, a metric called the Teichmüller metric is naturally induced by Teichmüller mappings. Mathematically, the Teichmüller metric is defined as follows.

Definition 6.3 (Teichmüller metric). *For every i, let M_i be a Riemann surface with feature landmarks $\{p_{ik}\}_{k=1}^n$, Ω be the template surface with the same topology as M_i with corresponding landmarks $\{q_k\}_{k=1}^n$. Suppose each surface M_i is parameterized onto Ω by a landmark-matching quasi-conformal homeomorphism $f_i : M_i \to \Omega$. The Teichmüller metric between (f_i, M_i) and (f_j, M_j) is defined as*

$$d_T((f_i, M_i), (f_j, M_j)) = \inf_{\varphi} \frac{1}{2} \log K(\varphi), \qquad (6.46)$$

where $\varphi : M_i \to M_j$ varies over all quasi-conformal mappings with $\{p_{ik}\}_{k=1}^n$ corresponds to $\{p_{jk}\}_{k=1}^n$, which is homotopic to $f_j^{-1} \circ f_i$, and K is the maximal quasi-conformal dilation.

According to the composition property of quasi-conformal mappings, neither the left nor the right composition with a conformal mapping affect the maximal quasi-conformal dilation K. Therefore, if we consider any $\varphi = g_j^{-1} \circ h \circ g_i$, where g_l is a conformal mapping from $\mathcal{M}_l$ to the rectangle $\tilde{R}_l$, it follows that

$$K(\varphi) = K(h). \qquad (6.47)$$

Thus, the Teichmüller metric is uniquely determined by the maximal quasi-conformal dilation of the extremal mapping between the two rectangular domains.

By Theorem 4.5, with a suitable boundary condition, the extremal mapping between two unit disks is Teichmüller and unique. A similar result also holds for the extremal mapping between two rectangular domains. Therefore, the infimum in Equation (6.46) is achieved by the unique Teichmüller mapping between $\tilde{R}_i$ and $\tilde{R}_j$. To be more specific, we have

$$d_T((f_i, M_i), (f_j, M_j)) = \frac{1}{2} \log K(h) \qquad (6.48)$$

where $h : \tilde{R}_i \to \tilde{R}_j$ is a landmark-constrained Teichmüller mapping with $g_j^{-1} \circ h \circ g_i$ homotopic to $f_j^{-1} \circ f_i$. Hence, the difference of two feature-endowed point clouds can be evaluated in terms of the quasi-conformal dilation of the Teichmüller mapping between them. This motivates us the following method for the shape analysis of feature-endowed point clouds.

Let $\{P_i\}$ be a collection of feature-endowed point clouds. To perform an accurate classification on $\{P_i\}$, we take a template point cloud surface P and compute the landmark-matching Teichmüller mappings $\mathbf{f}_i : P_i \to P$ for all i. Then, we evaluate the associated discrete diffuse PCBCs $\boldsymbol{\mu}_i$ and denote the Teichmüller distance of P_i and P by

$$d_i = \frac{1}{2} \log \frac{1 + \|\boldsymbol{\mu}_i\|_\infty}{1 - \|\boldsymbol{\mu}_i\|_\infty}. \tag{6.49}$$

The value d_i records the difference between the point cloud P_i and the template point cloud surface P. Specifically, if P_i and P_j are with similar shapes, then $|d_i - d_j|$ is small. By clustering these values d_i, we can effectively classify different feature-endowed point clouds. It is noteworthy that d_i are merely one-dimensional real values. Therefore, the shape analysis problem of 3D feature-endowed point clouds is significantly reduced to a problem between scalar values.

The classification power of our TEMPO method can be further enhanced by increasing the number of template point clouds. Instead of computing the landmark matching Teichmüller mappings from all point clouds to a single template, we can consider using N template point clouds, where $N > 1$. In this case, for each point cloud P_i, we obtain a N-dimensional vector

$$\vec{d}_i = (d_{i1}, d_{i2}, \cdots, d_{iN})^T \tag{6.50}$$

of Teichmüller distances, where d_{ij} is the Teichmüller distance between P_i and the template point cloud T_j. By computing the distances between the distance vectors d_i, we can obtain a more accurate classification result.

In the discrete case, numerical errors are unavoidable. To alleviate the effect of very minor outlying Beltrami coefficients, we replace $\|\mu\|_\infty$ by the average of $|\mu|$ in the calculation of the Teichmüller distances. The validity of this modification is justified by the negligible variance of $|\mu|$. Our algorithm for the shape analysis is summarized in Algorithm 4. Experimental results are provided in Section 7 to illustrate the effectiveness of our proposed method.

Algorithm 4: Representing point clouds as vectors in $\mathbb{R}^N$ using the Teichmüller metric

Input: A set of point clouds $\{P_i\}_{i=1}^M$, a set of template point clouds $\{T_j\}_{j=1}^N$, with the landmark correspondences $p_{i,l} \leftrightarrow q_{j,l}$, $p_{i,l} \in P_i$, $q_{j,l} \in T_j$, $l = 1, 2, ..., n$

Output: The distance vectors $\vec{d}_i$ in $\mathbb{R}^N$, $i = 1, 2, \cdots, M$

1 Using Algorithm 3, compute the Teichmüller mapping $\mathbf{f}_{ij}$ from each point cloud P_i to the template point cloud T_j with the landmark correspondences;

2 Compute the Teichmüller distance d_{ij} between P_i and T_j by
$d_{ij} = \frac{1}{2} \log(K(\mathbf{f}_{ij}))$;

3 The distance vector for P_i is given by $\vec{d}_i = (d_{i1}, d_{i2}, \cdots, d_{iN})^T$.

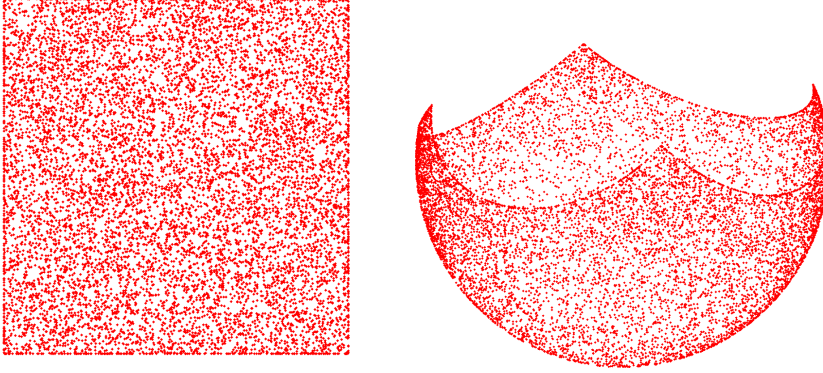---

FIG. 7.1.   *An example used in evaluating different numerical schemes for comput-*
ing conformal mappings on point clouds.   The conformal mapping is given by $f(x,y) =$
$\left(\dfrac{2x}{x^2 + y^2 + 1}, \dfrac{2y}{x^2 + y^2 + 1}, \dfrac{x^2 + y^2 - 1}{x^2 + y^2 + 1} \right).$

7. Experimental Results. In this section, we demonstrate the effectiveness of
our proposed TEMPO method by various examples. Our proposed algorithms are
implemented in MATLAB. The point clouds are adapted from the AIM@SHAPE
shape repository [51], the TF3DM repository [54], the Laboratory for Computational
Longitudinal Neuroimaging (LCLN) shape database [53], the human face dataset [52]
and the Spacetime Face Data [55]. Some additional facial point clouds are sampled
using Kinect for the shape analysis experiment. The sparse linear systems are solved
using the built-in backslash operator (\) in MATLAB. All experiments are performed
on a PC with an Intel(R) Core(TM) i7-4770 CPU @3.40 GHz processor and 16.00
GB RAM.

7.1. The performance of our proposed approximation schemes. In this
subsection, we evaluate the performance of our proposed approximation schemes for
computing quasi-conformal (including conformal) mappings on point clouds with disk
topology.

We first compare our proposed scheme with the local mesh method [24] and the
moving least square method with special weight [27, 28] for computing conformal
mappings. In each experiment, we first generate a random point cloud on a planar
rectangular domain. Then, we transform the point cloud using a conformal mapping
with an explicit formula. An example is given in Figure 7.1. On the transformed
point cloud, we approximate the Laplace-Beltrami operator using the aforementioned
schemes. Using the approximated Laplace-Beltrami operator, we solve the Laplace
equation to map the point cloud back onto the rectangular domain. The resulting
position errors show the accuracies of the approximation schemes. Table 7.1 lists
the statistics of several experiments. It can be observed that our proposed combined
scheme provides better approximations for the conformal mappings on point clouds
with disk topology.

Then, we compare the mentioned approximation schemes for computing quasi-
conformal mappings with prescribed Beltrami coefficients. This time, in each ex-
periment, we transform a randomly generated point cloud using a quasi-conformal

Example 1

Method	maximum position error	average 1-norm error	average 2-norm error
Local Mesh [24]	0.048	0.017	0.00031
MLS with special weight [27, 28]	1.821	0.275	0.13026
Our proposed method	0.048	0.017	0.00031

Example 2

Method	maximum position error	average 1-norm error	average 2-norm error
Local Mesh [24]	0.022	0.0043	0.000020
MLS with special weight [27, 28]	0.853	0.2529	0.071753
Our proposed method	0.011	0.0035	0.000014

Example 3

Method	maximum position error	average 1-norm error	average 2-norm error
Local Mesh [24]	0.011	0.0041	0.000017
MLS with special weight [27, 28]	0.182	0.0659	0.004212
Our proposed method	0.010	0.0035	0.000013

TABLE 7.1

The performances of different numerical schemes for computing conformal mappings on point clouds.

mapping with an explicit formula. An example is given in Figure 7.2. On the transformed point cloud, we approximate the generalized Laplacian using different schemes and solve the generalized Laplace equation to map the point cloud back onto the rectangular domain. Table 7.2 lists the statistics of several experiments. Again, our combined approximation scheme produces results with higher accuracy.

7.2. Landmark constrained Teichmüller Parameterizations. After demonstrating the advantage of our proposed approximation scheme for quasi-conformal mappings, we illustrate the effectiveness of our landmark constrained Teichmüller parameterization algorithm using numerous experiments. In all experiments, the stopping criterion is set to be $\epsilon = 10^{-6}$ and the initial balancing parameter γ in the hybrid equation (6.8) is set to be $\gamma = \frac{1}{2}$.

Figure 7.3 shows the landmark-constrained Teichmüller parameterization result of a twisted point cloud. The green points and the blue points represent the prescribed landmark constraints on the planar domain. Even under the large landmark deformations, our parameterization result ensures a uniform conformality distortion. This indicates the Teichmüller property of our proposed algorithm. More examples are shown in Figure 7.4 and Figure 7.5. It can be easily observed that in all of our experiments, the norms of the Beltrami coefficients always accumulate at a specific value. This indicates that our parameterization results of point clouds closely resemble the continuous Teichmüller parameterizations of simply-connected open surfaces.

Extensive experiments of our proposed landmark constrained Teichmüller parameterization algorithm have been carried out using different point clouds with disk topology. The statistics of the experiments are listed in Table 7.3. Our proposed method is highly efficient. The computations of the landmark-matching Teichmüller parameterizations complete within 1 minute on average for point clouds with moder-
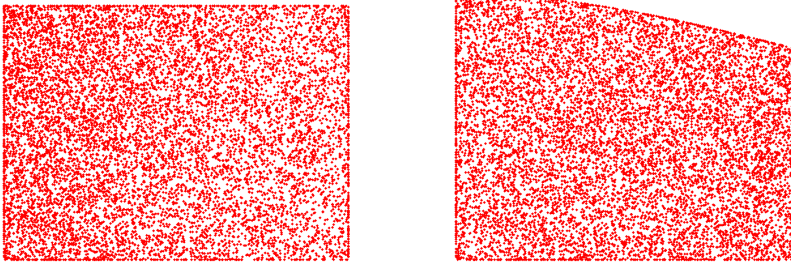
FIG. 7.2.   *An example used in evaluating different schemes for computing quasi-conformal mappings on point clouds. The quasi-conformal mapping is given by $f(x,y) = \left(\log(x+1), \arcsin\left(\frac{y}{2 + (\log(x+1))^2} \right) \right)$.*
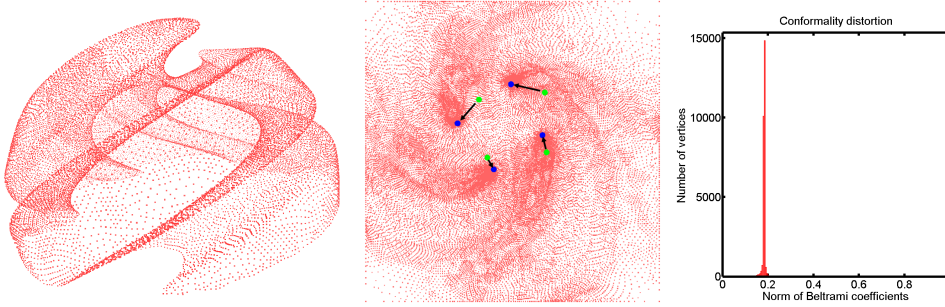

FIG. 7.3. *Teichmüller parameterization of a twisted point cloud. Left: The input point cloud. Middle: The parameterization result. The landmark constraints are represented by the green and blue points. The green points represent the original locations and the blue points are the target locations. Right: The histogram of the norm of the Beltrami coefficients. It can be observed that each histogram highly concentrates at one value. This shows that our parameterization achieves uniform conformality distortion.*

ate size. Even for point clouds with over 100K vertices, our algorithm only takes a few minutes. To quantitatively assess the performance of our proposed method, we compute the mean and the variance of the Beltrami coefficient $\boldsymbol{\mu}$ of every Teichmüller parameterization result. It is noteworthy that the variances of the Beltrami coefficients are close to zero in all experiments. In other words, the conformality distortions are uniform. As a remark, it can be observed in Table 7.3 that for denser point clouds, the variances of the results are in general smaller. All of the above results agree with the theories we have established. They reflect the accuracy of our proposed algorithm for Teichmüller parameterizations of feature-endowed point clouds.

7.3. Shape analysis of point clouds using Teichmüller metric. Using our proposed Teichmüller parameterization scheme, Teichmüller registrations between point clouds with prescribed landmark constraints can be efficiently computed as described in Algorithm 3. Figure 7.6 shows the registration of two human face point

Example 1

Method	maximum position error	average 1-norm error	average 2-norm error
Local Mesh [24]	0.0756	0.01150	0.00014248
MLS with special weight [27, 28]	0.0940	0.02524	0.00089952
Our proposed method	0.0252	0.00836	0.00006811

Example 2

Method	maximum position error	average 1-norm error	average 2-norm error
Local Mesh [24]	0.0467	0.01011	0.00010035
MLS with special weight [27, 28]	0.0212	0.01219	0.00012290
Our proposed method	0.0183	0.00750	0.00005012

Example 3

Method	maximum position error	average 1-norm error	average 2-norm error
Local Mesh [24]	0.0172	0.00813	0.00005617
MLS with special weight [27, 28]	0.6373	0.11550	0.02536965
Our proposed method	0.0172	0.00796	0.00005446

TABLE 7.2

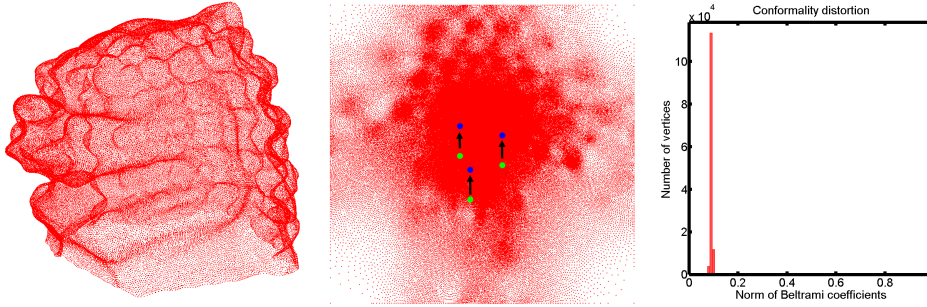*The performances of different schemes for computing quasi-conformal mappings on point clouds.*

FIG. 7.4. *Teichmüller parameterization of a lion point cloud. Left: The input point cloud. Middle: The parameterization result. The landmark constraints are represented by the green and blue points. The green points represent the original locations and the blue points are the target locations. Right: The histogram of the norm of the Beltrami coefficients.*

clouds by our proposed method. It can be observed that the landmark constraints are satisfied in the registration result. Moreover, the geometric features are optimally preserved under the registration. There is no unnatural distortion in any specific part of the registered point cloud.

Our proposed TEMPO algorithm can be applied for the shape analysis of point clouds with disk topology by making use of the Teichmüller metric. To demonstrate the effectiveness of our proposed method, we consider a face recognition problem of point clouds. Figure 7.7 shows a dataset of point clouds sampled from several human faces. For each subject, point clouds of four facial expressions (neutral, happy, sad, angry) are obtained. It is conceivable that even with different facial expressions, point clouds sampled from the same human should be similar in shape. In the set of all
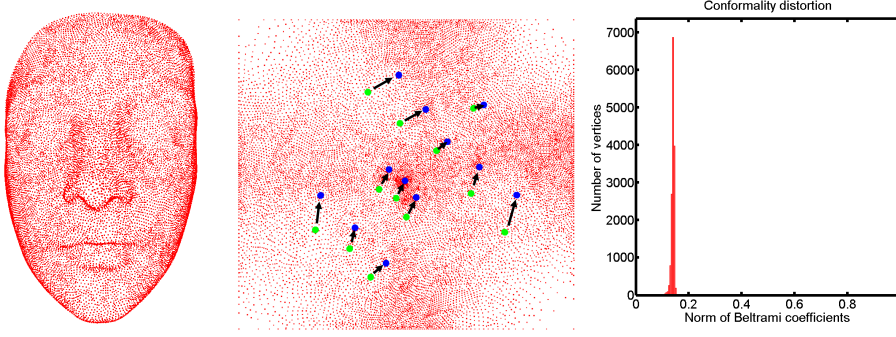
Fig. 7.5. *Teichmüller parameterization of Lucy point cloud. Left: The input point cloud. Middle: The parameterization result. The landmark constraints are represented by the green and blue points. The green points represent the original locations and the blue points are the target locations. Right: The histogram of the norm of the Beltrami coefficients.*

Point clouds	# of points	Time (m)	Mean($|\boldsymbol{\mu}|$)	Var($|\boldsymbol{\mu}|$)
Cesar	9755	0.3081	0.2402	6.1474e-04
Muscle guy	14240	0.3241	0.2934	9.8858e-04
Obese man	14415	0.3568	0.2508	7.7045e-04
Lucy	15167	0.6486	0.1523	1.4981e-04
Twisted	28031	1.1648	0.1986	8.0233e-04
Human face (neutral)	31350	0.6517	0.0713	3.3564e-05
Human face (angry)	31468	0.7428	0.0796	3.8812e-05
Human face (sad)	31543	0.6117	0.0693	5.6557e-05
Human face (happy)	31878	0.8536	0.0909	7.7383e-05
Bumps	114803	2.5353	0.1070	3.4902e-05
Lion	129957	2.5842	0.0893	2.8239e-05

TABLE 7.3
The performance of our landmark-constrained Teichmüller parameterization algorithm for various point clouds.

the sampled point clouds, we aim to correctly classify the point clouds into several groups, where each of the groups represents one human. 12 landmark constraints near the eyes, nose and mouth of each point cloud are labeled to ensure the accuracy of the classification.

To apply our TEMPO algorithm, we first select one human face (denoted by T1) in our dataset as a template surface. Then, for every facial point cloud, we first compute the conformal parameterization of them onto planar domains. Then, we compute the landmark-constrained Teichmüller registration between the planar parameterizations of every facial point cloud and the template point cloud. The registration results are recorded in Table 7.4.

In the registration results listed in in the columns of T1 in Table 7.4, we can notice the significance of the Teichmüller distances. More explicitly, point clouds of different facial expressions sampled from the same subject result in close Teichmüller distances. The results reflect the strength of our TEMPO algorithm for the shape analysis and classifications of feature-endowed point clouds.

To evaluate the classification accuracy achieved by our TEMPO method, we com-
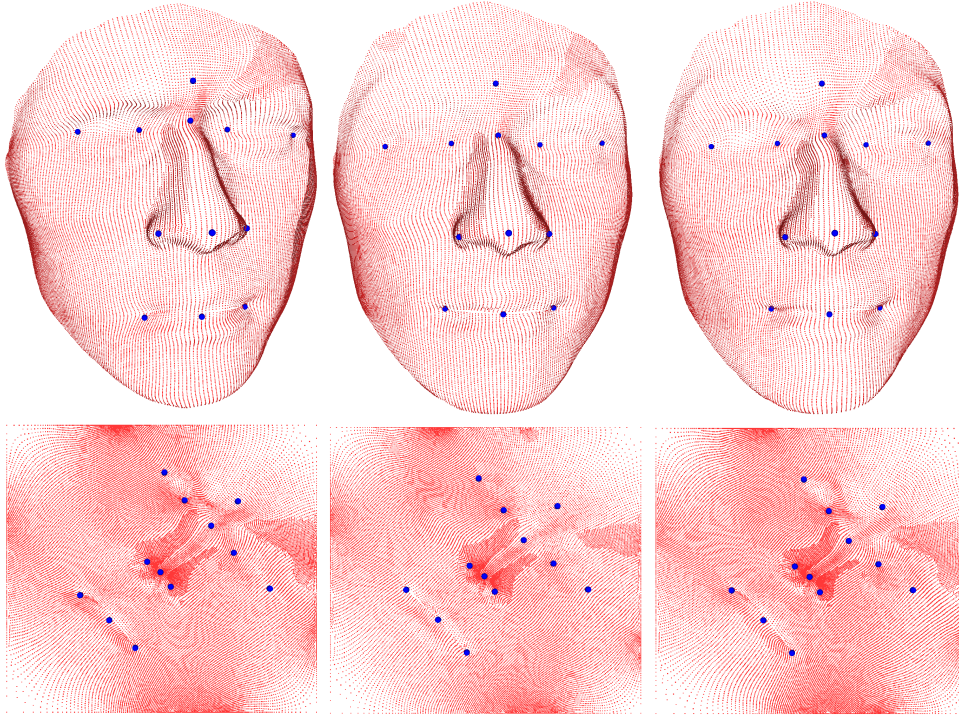
FIG. 7.6. *Teichmüller registration between two human face point clouds with prescribed landmark constraints. Top left: The source point cloud. Top middle: The target point cloud. Top right: The registration result by our algorithm. Bottom: The conformal parameterizations of the source and target point clouds, and the landmark constrained Teichmüller parameterization. The blue points refer to the facial landmark constraints.*

pare our method with the existing methods for point cloud mapping. For a fair comparison, we consider a feature-endowed point cloud mapping algorithm in [20]. In [20], İrfanoğlu et al. proposed an automatic registration scheme for feature-endowed facial point clouds. Using their proposed method, a distance is computed between every point cloud and the template. The results are listed in the rightmost columns of Table 7.4. The distances are plotted in Figure 7.8. In contrast to our Teichmüller distances, the distances computed by [20] could not distinguish the point clouds sampled from different subjects.

We now consider using more template point clouds for further enhancing the classification accuracy. Besides using the template point cloud T1, we repeat the above experiments for another template point cloud (denoted by T2) and record the results in the columns of T2 in Table 7.4. Figure 7.9 shows the classification results using both of the two template point clouds T1 and T2. In this case, each point cloud in the dataset is represented by a point in $\mathbb{R}^2$, where the x-axis and the y-axis respectively correspond to the distances with respect to T1 and T2. Again, the Teichmüller metric leads to a better classification result when compared with the method in [20]. The comparisons demonstrate the power of our TEMPO method in shape analysis of point clouds.

8. Conclusion and Future Work. In this paper, we have developed the notion of PCT-maps, a discrete analogue of the Teichmüller mappings on point clouds.

FIG. 7.7. *A dataset of human face point clouds with different facial expressions. Each row represents the facial expressions sampled from one human.*

The consistency between the PCT-maps and the continuous T-maps has been rigorously established. Based on the notion of PCT-maps, we have proposed a novel method called TEMPO for computing landmark-matching Teichmüller mappings of point clouds with disk topology. Firstly, we have introduced a hybrid quasi-conformal reconstruction scheme for computing quasi-conformal mappings on point clouds. Secondly, an improved method has been proposed for approximating differential opera-
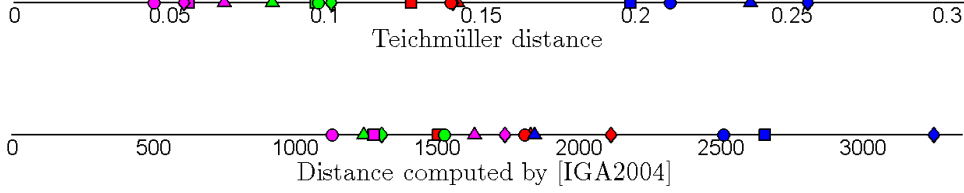
FIG. 7.8. *Classification of the face data using our Teichmüller distance and the distance computed by [20]. Each color represents one human (red: subject 1, green: subject 2, blue: subject 3, purple: subject 4), and each shape represents a type of facial expressions (circle: neutral, square: happy, triangle: sad, diamond: angry).*
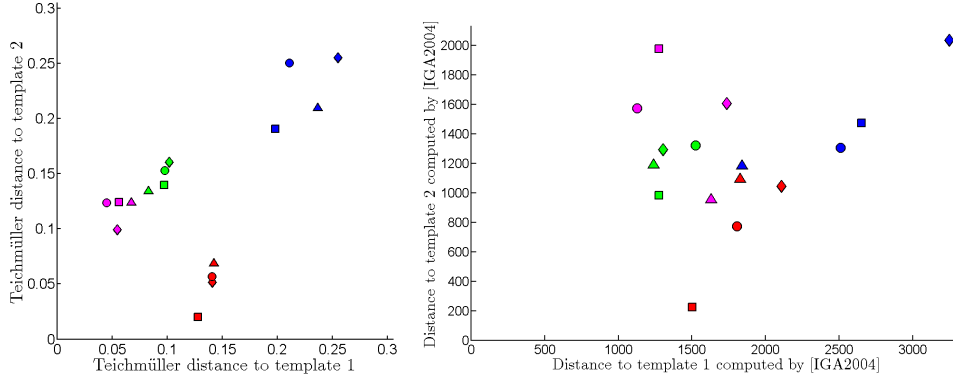

FIG. 7.9. *Classification of the face data using our Teichmüller distance and the distance computed by [20] with respect to two template point clouds.*

tors on point clouds with disk topology. The new approximation method produces more accurate results. Thirdly, we have presented an algorithm for computing the Teichmüller parameterizations of disk-type point clouds with rigorous mathematical guarantees. Unlike the conventional approaches, our method allows prescribed landmark constraints on the point clouds and guarantees uniform conformality distortions. Fourthly, the induced Teichmüller metric enables us to accurately classify different feature-endowed point clouds. Experimental results have demonstrated the effectiveness of our proposed TEMPO algorithm. In the future, we aim to extend our Teichmüller parameterization and registration methods for point clouds with arbitrary topologies.

REFERENCES

[1] P. Breitkopf, H. Naceur, A. Rassineux, and Pierre Villon, *Moving least squares response surface approximation: Formulation and metal forming applications.* Computers & Structures, Volume 83, Issues 17–18, pp. 1411–1428, 2005.

[2] 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, Volume 8, Issue 1, pp. 67–94, 2015.

[3] P. T. Choi and L. M. Lui, *Fast disk conformal parameterization of simply-connected open surfaces.* Journal of Scientific Computing, Volume 65, Issue 3, pp. 1065–1090, 2015.

Point cloud	Registration time (m)		Teichmüller distance		Distance by [20]	
	T1	T2	T1	T2	T1	T2
Subject 1 (neutral)	0.6012	0.3366	0.1278	0.0202	1500.6	225.8
Subject 1 (happy)	0.6283	0.5853	0.1426	0.0684	1829.5	1094.6
Subject 1 (sad)	0.6284	0.5177	0.1412	0.0515	2110.8	1043.6
Subject 1 (angry)	0.6307	0.5280	0.1405	0.0566	1806.2	774.1
Subject 2 (neutral)	0.4297	0.3910	0.0971	0.1397	1274.9	984.2
Subject 2 (happy)	0.4471	0.4881	0.0831	0.1339	1241.3	1190.9
Subject 2 (sad)	0.4660	0.4858	0.1021	0.1605	1305.0	1292.8
Subject 2 (angry)	0.4296	0.4754	0.0978	0.1531	1524.3	1322.6
Subject 3 (neutral)	0.4013	0.4939	0.1981	0.1906	2651.8	1473.1
Subject 3 (happy)	1.0897	0.6797	0.2368	0.2094	1841.2	1182.7
Subject 3 (sad)	0.3549	0.5171	0.2522	0.2553	3249.7	2035.1
Subject 3 (angry)	0.3965	0.5093	0.2108	0.2503	2508.8	1306.9
Subject 4 (neutral)	0.3454	0.4128	0.0562	0.1242	1275.6	1977.1
Subject 4 (happy)	0.3678	0.5134	0.0677	0.1239	1630.6	953.5
Subject 4 (sad)	0.2841	0.3961	0.0548	0.0994	1738.3	1607.4
Subject 4 (angry)	0.3338	0.4303	0.0450	0.1235	1128.9	1574.2

TABLE 7.4

Shape analysis of feature-endowed point clouds using our proposed TEMPO method and the method in [20]. In each experiment, every point cloud is registered with a common template point cloud surface (T1 or T2) by a landmark-matching Teichmüller mapping of their planar conformal parameterizations. The resulting Teichmüller distance between each point cloud and the template point cloud provides us with an accurate measure of the similarity of the point clouds.

[4] P. T. Choi and L. M. Lui, *A linear algorithm for disk conformal parameterization of simply-connected open surfaces*. Preprint, arXiv:1508.00396, 2015.

[5] P. T. Choi, K. T. Ho, and L. M. Lui, *Spherical conformal parameterization of genus-0 point clouds for meshing*. Preprint, arXiv:1508.07569, 2015.

[6] M. Desbrun, M. Meyer, and P. Alliez, *Intrinsic parameterizations of surface meshes*, Computer Graphics Forum, Volume 21, pp. 209–218, 2002.

[7] M. Floater, *Parameterization and smooth approximation of surface triangulations*, Computer Aided Geometric Design 1997, Volume 14, Number 3, pp. 231–250, 1997.

[8] M. S. Floater, *Meshless parameterization and B-spline surface approximation*. Proceedings of the Ninth IMA Conference on the Mathematics of Surfaces, pp. 1–18, 2000.

[9] M. S. Floater and M. Reimers, *Meshless parameterization and surface reconstruction*. Computer Aided Geometric Design, Volume 18, Issue 2, pp. 77–92, 2001.

[10] M. Floater and K. Hormann, *Parameterization of triangulations and unorganized points*, Tutorials on Multiresolution in Geometric Modeling, 2002.

[11] M. Floater, *Mean value coordinates*, Computer Aided Geometric Design, Volume 20, Number 1, pp. 19–27, 2003.

[12] M. Floater and K. Hormann, *Surface parameterization: A tutorial and survey*, in Advances in Multiresolution for Geometric Modeling, pp. 157–186, 2005.

[13] F. Gardiner and N. Lakic. *Quasiconformal Teichmüller theory*. American Mathematics Society, 2000.

[14] X. Gu and S.T. Yau, *Computing conformal structures of surfaces*, Communications in Information and Systems, 2(2):121–146, 2002.

[15] X. Gu and S. T. Yau, *Global conformal surface parameterization*, Eurographics Symposium on Geometry Processing, 127–137, 2003.

[16] X. Gu, Y. Wang, T. F. Chan, P. M. Thompson, and S.-T. Yau, *Genus zero surface conformal mapping and its application to brain surface mapping*, IEEE Transactions on Medical Imaging, 23:949–958, 2004.

[17] X. Guo, X. Li, Y. Bao, X. Gu, H. Qin, *Meshless thin-shell simulation based on global conformal parameterization*. IEEE Transactions on Visualization and Computer Graphics, Volume 12, Issue 3, pp.375–385, 2006.

[18] K. Hormann and G. Greiner, *MIPS: An efficient global parametrization method*, in Curve and Surface Design, pp. 153–162, 2000.

[19] K. Hormann, B. Lévy, and A. Sheffer, *Mesh parameterization: Theory and practice*, SIGGRAPH 2007 Course Notes, 2:1–122, 2007.

[20] M. O. İrfanoğlu, B. Gökberk, and L. Akarun, *3D shape-based face recognition using automatically registered facial surfaces*. Proceedings of the 17th International Conference on Pattern Recognition, Volume 4, pp. 183–186, 2004.

[21] M. Jin, Y. Wang, X. Gu, and S. T. Yau, *Optimal global conformal surface parameterization for visualization*, Communications in Information and Systems, Volume 4, Number 2, pp. 117–134, 2005.

[22] M. Jin, J. Kim, F. Luo, and X. Gu, *Discrete surface Ricci flow*, IEEE Transaction on Visualization and Computer Graphics, 14(5), 1030–1043, 2008.

[23] L. Kharevych, B. Springborn, and P. Schröder, *Discrete conformal mappings via circle patterns*, ACM Transactions on Graphics, Volume 25, Number 2, pp. 412–438, 2006.

[24] R. Lai, J. Liang, and H. Zhao, *A local mesh method for solving PDEs on point clouds*. Inverse Problem and Imaging, Volume 7, Issue 3, pp. 737–755, 2013.

[25] D. Levin, *The approximation power of moving least-squares*. Mathematics of Computation, Volume 67, Issue 224, pp. 1517–1531, 1998.

[26] B. Lévy, S. Petitjean, N. Ray, and J. Maillot, *Least squares conformal maps for automatic texture atlas generation*, in ACM SIGGRAPH 2002, pp. 362–371, 2002.

[27] J. Liang, R. Lai, A. Wong and H. Zhao, *Geometric understanding of point clouds using Laplace-Beltrami operator*, IEEE Conference on Computer Vision and Pattern Recognition, pp. 214–221, 2012.

[28] J. Liang and H. Zhao, *Solving partial differential equations on point clouds*, SIAM Journal of Scientific Computing, Volume 35, Issue 3, pp. 1461–1486, 2013.

[29] Y. Lipman, *Bounded distortion mapping spaces for triangular meshes*. ACM Transactions on Graphics (SIGGRAPH), Volume 31, Issue 4, 108, 2012.

[30] W.K. Liu, S. Li and T. Belytschko, *Moving least-square reproducing kernel methods (I) methodology and convergence*, Computer Methods in Applied Mechanics and Engineering, Volume 143, pp. 113–154, 1997.

[31] L. M. Lui, K. C. Lam, T. W. Wong, and X. Gu, *Texture map and video compression using Beltrami representation*. SIAM Journal on Imaging Sciences, 6, pp.1880–1902, 2013.

[32] L. M. Lui, K. C. Lam, S. T. Yau, and X. Gu, *Teichmüller mapping (T-map) and its applications to landmark matching*. SIAM Journal on Imaging Sciences, Volume 7, Number 1, pp. 391–426, 2014.

[33] L. M. Lui, X. Gu, and S.-T. Yau, *Convergence of an Iterative Algorithm for Teichmüller Maps via Harmonic Energy Optimization*. Mathematics of Computation, Volume 84, Number 296, pp. 2823–2842, 2015.

[34] F. Luo, *Combinatorial Yamabe flow on surfaces*, Communications in Contemporary Mathematics, 6:765–780, 2004.

[35] T. W. Meng and L. M. Lui, *The theory of computational quasi-conformal geometry on point clouds*. Preprint, arXiv:1510.05104, 2015.

[36] D. Mirzaei, R. Schaback, M. Dehghan, *On generalized moving least squares and diffuse derivatives*. IMA Journal of Numerical Analysis, 32, pp. 983–1000, 2012.

[37] D. Mirzaei, *Analysis of moving least squares approximation revisited*. Journal of Computational and Applied Mathematics, Volume 282, pp. 237–250, 2015.

[38] P. Mullen, Y. Tong, P. Alliez and M. Desbrun, *Spectral conformal parameterization*. Computer Graphics Forum, Volume 27, Issue 5, 1487–1494, 2008.

[39] A. Nealen, *An as-short-as-possible introduction to the least squares, weighted least squares and moving least squares methods for scattered Data approximation and interpolation*, Technical Report, TU Darmstadt, 2004.

[40] R. Schoen and S. T. Yau, *Lectures on differential geometry*, International Press of Boston, Boston, MA, 1994.

[41] A. Sheffer and E. de Sturler, *Surface parameterization for meshing by triangulation flattening*, Proceedings of 9th International Meshing Roundtable, pp. 161–172, 2000.

[42] A. Sheffer, B. Lévy, M. Mogilnitsky, and A. Bogomyakov, *ABF++: Fast and robust angle based flattening*, ACM Transactions on Graphics, Volume 24, Number 2, pp. 311–330, 2005.

[43] A. Sheffer, E. Praun and K. Rose, *Mesh parameterization methods and their applications*, Foundations and Trends in Computer Graphics and Vision, 2(2):105–171, 2006.

[44] O. Weber, A. Myles, D. Zorin, *Computing extremal quasiconformal maps*. Computer Graphics Forum, 31(5), pp. 1679–1689, 2012.

[45] H. Wendland, *Piecewise polynomial, positive definite and compactly supported radial functions*

of minimal degree. Advances in Computational Mathematics, Volume 4, Issue 1, pp. 389–396, 1995.

[46] H. Wendland, *Local polynomial reproduction and moving least squares approximation.* IMA Journal of Numerical Analysis, Volume 21, Issue 1, pp. 285–300, 2001.

[47] H. Wendland, *Scattered data approximation.* Cambridge University Press, Vol.17, 2004.

[48] Y. L. Yang, R. Guo, F. Luo, S. M. Hu, and X. F. Gu, *Generalized discrete Ricci flow*, Computer Graphics Forum, 28(7):2005–2014, 2009.

[49] L. Zhang, L. Liu, C. Gotsman, and H. Huang, *Mesh reconstruction by meshless denoising and parameterization.* Computers and Graphics, Volume 34, Issue 3, pp. 198–208, 2010.

[50] M. Zhang, R. Guo, W. Zeng, F. Luo, S. T. Yau and X. Gu, *The unified discrete surface Ricci flow.* Journal of Graphical Models, 76(5): 321–339, 2014.

[51] AIM@SHAPE Shape Repository. http://visionair.ge.imati.cnr.it/ontologies/shapes/

[52] A. M. Bronstein, M. M. Bronstein, R. Kimmel, *Calculus of non-rigid surfaces for geometry and texture manipulation.* IEEE Trans. Visualization and Computer Graphics, Volume 13/5, pp. 902–913, 2007.

[53] Laboratory for Computational Longitudinal Neuroimaging (LCLN) Shape Database. http://reuter.mit.edu/research/shapeanalysis/shapedb1/

[54] TF3DM. http://tf3dm.com/

[55] L. Zhang, N. Snavely, B. Curless, and S. Seitz, *Spacetime Faces: High-resolution capture for modeling and animation.* ACM Transactions on Graphics (SIGGRAPH), Volume 23, Issue 3, pp. 548–558, 2004.