TEMPO: Feature-Endowed Teichmüller Extremal Mappings of Point Clouds[*]

Ting Wei Meng[†], Gary Pui-Tung Choi[‡], and Lok Ming Lui[†]

Abstract. In recent decades, the use of three-dimensional point clouds has been widespread in the computer industry. The development of techniques for 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 guarantees 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 recognition and classification 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

AMS subject classifications. 30C30, 49M25, 65K10

DOI. 10.1137/15M1049117

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 three-dimensional (3D) facial recognition and object classification. 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 registration between two feature-endowed point clouds is 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 shapes of infinitesimally small figures. 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*, is particularly

[†]Department of Mathematics, The Chinese University of Hong Kong Shatin, Hong Kong (twmeng@math.cuhk.edu.hk, lmlui@math.cuhk.edu.hk).
[‡]John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 (pchoi@g.harvard.edu).

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 arising from the computation of conformal and quasi-conformal mappings on point clouds with disk topology. With the abovementioned schemes, we propose an efficient method for computing the Teichmüller Extremal Mapping of POint 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 the prescribed 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 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 squares conformal maps (LSCM). The parameterization method is

based on a least-squares approximation of the Cauchy–Riemann equations. Desbrun, Meyer, and Alliez [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, Springborn, and Schröder [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 to 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, Myles, and Zorin [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 an 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 [8] and Floater and Reimers [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 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, Ho, and Lui [5] proposed the north-south reiterations for computing the spherical conformal parameterizations

Table 1
A comparison of the parameterization schemes for point clouds with disk topology.

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

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 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 point cloud Teichmüller mapping (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 to point cloud surfaces.
2. We introduce an improved method for approximating differential operators arising from the computation of conformal and quasi-conformal mappings on point clouds with disk topology by combining the moving least squares (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. The induced Teichmüller metric serves as an effective dissimilarity metric for accurate classification of point clouds.
7. Besides the computational advantages, our TEMPO method is theoretically supported. Under proper assumptions on the input point cloud, 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*

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

The mentioned scalar function λ is called the *conformal factor*. It follows that every conformal mapping preserves angles and hence the shapes at an infinitesimal scale. 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, or
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 analysis. 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 are 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*

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

for some complex-valued function μ satisfying $||\mu||_\infty < 1$ and that $\frac{\partial f}{\partial z}$ is nonvanishing almost everywhere.
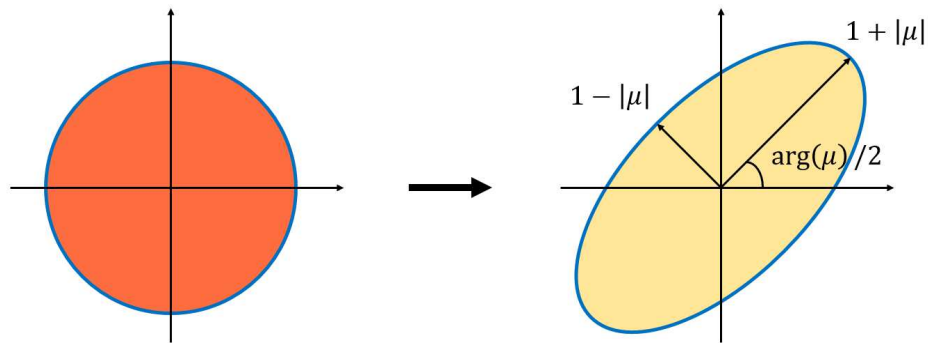
Figure 1. *A geometric illustration of quasi-conformal mappings. The information about the maximal magnification and shrinkage can be determined by the associated Beltrami coefficient μ.*

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 nonconformality of f. It measures how far the mapping f at each point deviates from a conformal mapping. A geometric illustration of quasi-conformal mappings is provided in Figure 1. In addition, the *maximal quasi-conformal dilation* of f is given by

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

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 4.3. *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*

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

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.

More explicitly, let μ be a given Beltrami coefficient associated with a quasi-conformal map $f : \mathbb{C} \to \mathbb{C}$. The computation of f can be achieved by solving an equation analogous to the Laplace equation $\Delta f = 0$. The method is outlined as follows.

Denote $f = u + iv$ and $\mu = \rho + i\tau$. By considering the Beltrami equation (4.2), each pair of the partial derivatives v_x, v_y and u_x, u_y can be expressed as linear combinations of the other [31],

$$(4.5) \qquad \begin{aligned} v_y &= \alpha_1 u_x + \alpha_2 u_y, \\ -v_x &= \alpha_2 u_x + \alpha_3 u_y, \end{aligned} \quad \text{and} \quad \begin{aligned} -u_y &= \alpha_1 v_x + \alpha_2 v_y, \\ u_x &= \alpha_2 v_x + \alpha_3 v_y, \end{aligned}$$

where $\alpha_1 = \frac{(\rho-1)^2 + \tau^2}{1 - \rho^2 - \tau^2}$, $\alpha_2 = -\frac{2\tau}{1 - \rho^2 - \tau^2}$, $\alpha_3 = \frac{(1+\rho)^2 + \tau^2}{1 - \rho^2 - \tau^2}$. Since $\nabla \cdot \left(\begin{smallmatrix} -v_y \\ v_x \end{smallmatrix} \right) = 0$ and $\nabla \cdot \left(\begin{smallmatrix} -u_y \\ u_x \end{smallmatrix} \right) = 0$, f can be computed by solving the following equations:

$$(4.6) \qquad \nabla \cdot \left(A \begin{pmatrix} u_x \\ u_y \end{pmatrix} \right) = 0 \ \text{ and } \ \nabla \cdot \left(A \begin{pmatrix} v_x \\ v_y \end{pmatrix} \right) = 0,$$

where $A = \left(\begin{smallmatrix} \alpha_1 & \alpha_2 \\ \alpha_2 & \alpha_3 \end{smallmatrix} \right)$. We call (4.6) the *generalized Laplace equation*. It can be observed that if $\mu = 0$, then (4.6) becomes the usual Laplace equation for solving conformal maps. In the discrete case, the above elliptic PDEs (4.6) can be discretized into sparse symmetric positive definite linear systems as described in [31, 2].

The analogy between the Laplace equation and the generalized Laplace equation can be interpreted in terms of the Riemannian metric. Given a quasi-conformal mapping $(f, |dz|^2)$ associated with the Beltrami coefficient μ, we can then 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. This can be explained by the following proposition.

Proposition 4.4 (see [5, page 13]). *Let $(S_1, \sigma |dz|^2)$ and $(S_2, \rho |dw|^2)$ be two domains on $\mathbb{C}$ with the metric tensors $\sigma |dz|^2$ and $\rho |dw|^2$, and let μ be a prescribed Beltrami coefficient on S_1 associated with a quasi-conformal map $f : (S_1, \sigma |dz|^2) \to (S_2, \rho |dw|^2)$. Then, the map solved by the generalized Laplace equation (4.6) is a harmonic map between $(S_1, |dz + \mu d\bar{z}|^2)$ and $(S_2, \rho |dw|^2)$. Consequently, solving the generalized Laplace equation (4.6) is equivalent to solving the Laplace equation with respect to the distorted metric tensor $|dz + \mu d\bar{z}|^2$.*

Finally, the abovementioned definitions and concepts of quasi-conformal maps between planar domains can be naturally extended onto Riemann surfaces by introducing the concept of *Beltrami differential*. A Beltrami differential $\mu(z) \frac{d\bar{z}}{dz}$ on a Riemann surface S is an assignment to each chart (U_α, ϕ_α) of an L_∞ complex-valued function μ_α, defined on local parameter z_α such that $\mu_\alpha \frac{d\bar{z_\alpha}}{dz_\alpha} = \mu_\beta \frac{d\bar{z_\beta}}{dz_\beta}$, on the domain which is also covered by another chart (U_β, ϕ_β). Here, $\frac{dz_\beta}{dz_\alpha} = \frac{d}{dz_\alpha} \phi_{\alpha\beta}$ and $\phi_{\alpha\beta} = \phi_\beta \circ \phi_\alpha$. An orientation preserving diffeomorphism $f : M \to N$ is said to be *quasi-conformally* associated with $\mu(z) \frac{d\bar{z}}{dz}$ if for any chart (U_α, ϕ_α) on M and any chart (U_β, ψ_β) on N, the mapping $f_{\alpha\beta} := \psi_\beta \circ f \circ \phi_\alpha^{-1}$ is quasi-conformally associated with $\mu_\alpha \frac{d\bar{z_\alpha}}{dz_\alpha}$.

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 mapping, called Teichmüller mappings, which produce uniform conformality distortions with prescribed landmark constraints. Mathematically, the Teichmüller mappings are defined as follows.

Definition 4.5 (Teichmüller mapping). *Let $f : M \to N$ be a quasi-conformal mapping. f is said to be a* Teichmüller mapping *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*

$$(4.7) \qquad \mu(f) = k \frac{\overline{\varphi}}{|\varphi|}$$

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 4.6 (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*

$$(4.8) \qquad\qquad\qquad K(f) \leq K(h),$$

where $K(f)$ is the maximal quasi-conformal dilation of f. It is uniquely extremal if the inequality (4.8) is strict when $h \neq f$.

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

Theorem 4.7 (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 exists a unique 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. Here $(D, \{p_i\}_{i=1}^n)$ denotes the unit disk D with prescribed landmark points $\{p_i\}_{i=1}^n$.*

Therefore, besides being 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.

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 4.8 (interior cone condition). *A domain $\Omega \subset \mathbb{R}^2$ is said to* satisfy the 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

$$(4.9) \qquad C(x, d, \theta, r) = \{x + ty : y \in \mathbb{S}^1, \, y^T d(x) \geq \cos \theta, \, 0 \leq t \leq r\}.$$

Definition 4.9 (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*

$$(4.10) \qquad h_{P,\Omega} := \sup_{x \in \Omega} \min_{p \in P} \|x - p\|$$

and

$$(4.11) \qquad q_P := \frac{1}{2} \min_{\substack{p_1, p_2 \in P \\ p_1 \neq p_2}} \|p_1 - p_2\|.$$

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

$$(4.12) \qquad q_P \leq h_{P,\Omega} \leq c_{qu} q_P.$$

From now on, we assume that all planar point clouds are quasi-uniform and are sampled from domains which satisfy the 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.

4.5. The MLS method for approximating differential operators on point clouds. One common approach to approximate functions and derivatives on point clouds is the *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].

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

$$(4.13) \qquad q(x_1, x_2) = [1, x_1, x_2, x_1^2, x_1 x_2, x_2^2]^T,$$

$$(4.14) \qquad q_1(x_1, x_2) = \partial_1 q = [0, 1, 0, 2x_1, x_2, 0]^T,$$

and

$$(4.15) \qquad q_2(x_1, x_2) = \partial_2 q = [0, 0, 1, 0, x_1, 2x_2]^T.$$

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

$$(4.16) \qquad c_x = argmin_c \sum_{i=1}^{N} w(\|x - p_i\|) \left(c^T q(p_i) - f_i\right)^2.$$

Table 2

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

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

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

$$(4.17) \qquad c_x = (Q^T W(x) Q)^{-1} Q^T W(x) F,$$

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 ith 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 2. Denote

$$(4.18) \qquad A_x = (Q^T W(x) Q)^{-1} Q^T W(x).$$

Then, the function is locally approximated by

$$(4.19) \qquad \tilde{f}(y) = q^T(y) A_x F.$$

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

The following error estimates for MLS have been established in [36].

Proposition 4.10 (see Corollary 4.10 in [36]).

$$(4.20) \qquad \|q_j^T(x) A_x\|_1 = O(h^{-1}),$$

where h is the fill distance.

Proposition 4.11 (see Theorem 4.3 in [36]).

$$\|\partial_i \partial_j f - (\partial_i \partial_j q^T(y)) A_x f\| = O(h), \tag{4.21}$$

where h is the fill distance.

4.6. Quasi-conformal geometry on point clouds. 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 4.12 (discrete diffuse PCBC [35]). *Given a point cloud $P = \{p_i \in \mathbb{R}^2 : i = 1, \ldots, N\}$ 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 PCBC $\tilde{\mu} : \Omega \to \mathbb{C}$ *is defined by*

$$\tilde{\mu}(x) = \frac{\begin{bmatrix} q_1^T(x) & q_2^T(x) \end{bmatrix} \begin{bmatrix} A_x & iA_x \\ iA_x & -A_x \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix}}{\begin{bmatrix} q_1^T(x) & q_2^T(x) \end{bmatrix} \begin{bmatrix} A_x & iA_x \\ -iA_x & A_x \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix}}, \tag{4.22}$$

where q_1, q_2, and A_x are, respectively, defined as in (4.14), (4.15) *and* (4.18). *In addition, the discrete diffuse PCBC is a complex-valued point cloud function defined by $\tilde{\boldsymbol{\mu}} = \tilde{\mu}|_P$.*

The following proposition provides an error estimate for the discrete diffuse PCBC.

Proposition 4.13 (see [35, page 9]). *Let P be a point cloud sampled from Ω with fill distance h, and $\mathbf{f}$ be the point cloud map corresponding to f. Denote the Beltrami coefficient associated with f by μ, and the discrete diffuse PCBC associated with $\mathbf{f}$ by $\tilde{\boldsymbol{\mu}}$. Then there exist some constants $C(f)$ such that if $h \leq C(f)$, we have the following error bound:*

$$|\mu(x) - \tilde{\boldsymbol{\mu}}(x)| \leq C(f)h^2. \tag{4.23}$$

Note that the abovementioned concepts are defined on planar domains. To extend the concept of quasi-conformal point cloud mappings on planar domains to the general 3D case, Meng and Lui [35] introduced a concept called ϵ-conformal parameterization for 3D point clouds, which is an analogue of conformal parameterization for 3D surfaces. Then, quasi-conformal point cloud mappings can be defined as a composition of ϵ-conformal parameterizations and planar quasi-conformal mappings. Readers are referred to [35] for more details.

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 PCT-map).   *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 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,*

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

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 ith row of L.

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

Lemma 5.2. *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

$$(5.2) \qquad \begin{aligned} f_x u + u_x f &= f_y v + v_y f, \\ f_y u + u_y f &= -f_x v - v_x f. \end{aligned}$$

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

$$(5.3) \qquad \begin{aligned} f_x &= G_1 f, \\ f_y &= G_2 f, \end{aligned}$$

where

$$(5.4) \qquad \begin{aligned} G_1 &= -uu_x - vv_x - vu_y + uv_y, \\ G_2 &= -uu_y - vv_y - uv_x + vu_x. \end{aligned}$$

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

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

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

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

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

$$(5.7) \qquad \partial_y G_1 - \partial_x G_2 = u\Delta v - v\Delta u = 0.$$

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

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

Consequently, the solution f is given by

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

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. ■

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 Teichmüller mappings.

Proposition 5.3. *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 $g|_{P_n}$, and $\tilde{\boldsymbol{\nu}}_n(p)$ be its value at a point p. We have

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

By Proposition 4.13, we have $|\mu(p) - \tilde{\boldsymbol{\nu}}_n(p)| \leq C(g)h_n^2$. 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:

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

Then we have

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

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

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

Therefore, when n is large enough,

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

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 $k \neq 0$. Now, we fix a point cloud P_n and consider the matrix L which is the approximation of the Laplace–Beltrami operator. By (4.21), for any interior point $p = p_i \in P_n$,

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

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

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

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 a direct calculation,

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

By taking $u = \rho$, and $v = -\tau$ in Lemma 5.2, 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. ∎

From the above proposition, 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 Teichmüller mappings. Then, the next step is to define the Teichmüller mappings between general disk-type point cloud surfaces.

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. Then we

define $\mathbf{f}$ to be a *PCT-map* between the two point cloud surfaces if $\phi_2^{-1} \circ \mathbf{f} \circ \phi_1$ is a planar PCT-map.

Using the above definition, a property analogous to Proposition 5.3 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. The pipeline of our proposed TEMPO algorithm is shown in Figure 2. It can be observed that our TEMPO algorithm involves computing a number of mappings. Hence, it is necessary to first establish effective numerical schemes for computing different mappings. Then, we can introduce our proposed TEMPO algorithm and finally discuss its application in shape analysis.

This section primarily consists of four subsections:

1. A hybrid quasi-conformal reconstruction scheme for computing quasi-conformal mappings on point clouds.
2. An improved approximation of the differential operators arising from the computation of conformal and quasi-conformal mappings on point clouds with disk topology.
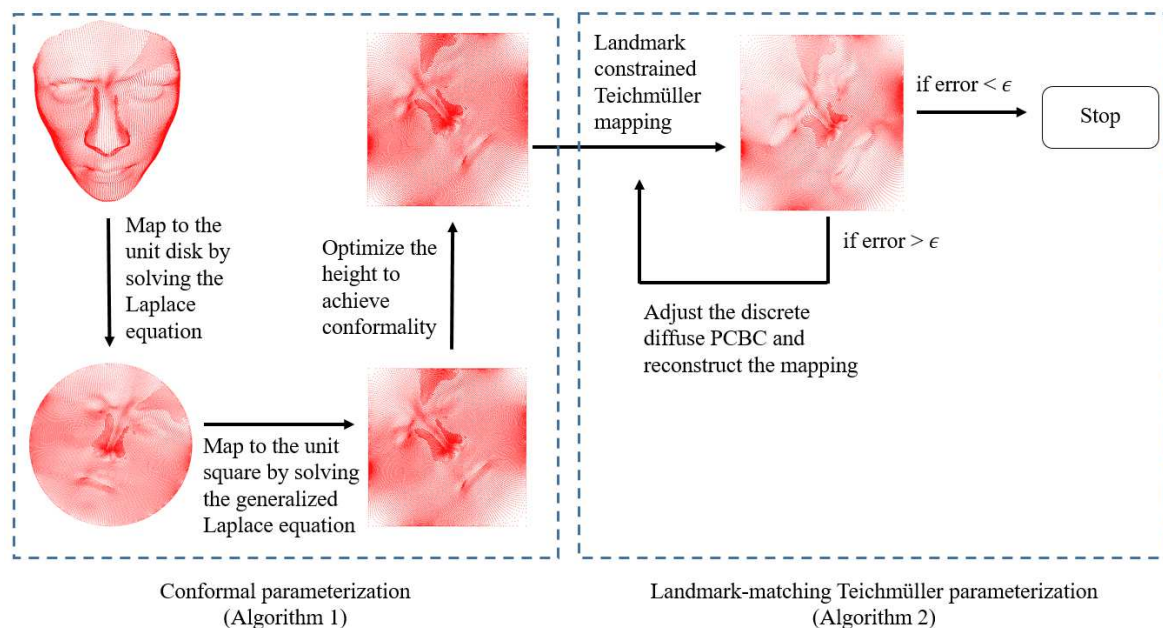

Figure 2. *The pipeline of our TEMPO algorithm.*

3. The algorithmic details of the TEMPO algorithm.

4. The induced Teichmüller metric for shape analysis of disk-type point clouds.

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 hybrid scheme for computing quasi-conformal mappings on point clouds with a prescribed 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 solve the linear system (4.5). Another approach for computing quasi-conformal mappings is the *linear Beltrami solver* proposed by Lui et al. in [31], which aims to solve the generalized Laplace equation (4.6).

The abovementioned methods work well on triangular meshes. However, for the case of point clouds, solely solving the system (4.5) may not result in any meaningful solutions. More specifically, the solution to the system (4.5) 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 (4.5) may not lead to a quasi-conformal diffeomorphism for point clouds. Also, the solution to the system (4.6) may be associated with a PCBC which is largely different from the input Beltrami coefficients. In other words, the accuracy of the system (4.6) 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:

$$(6.1) \qquad \begin{pmatrix} \alpha_1\partial_x + \alpha_2\partial_y & -\partial_y \\ \alpha_2\partial_x + \alpha_3\partial_y & \partial_x \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + \gamma \begin{pmatrix} \mathcal{L} & O \\ O & \mathcal{L} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = 0,$$

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

To solve the hybrid PDE (6.1) on point clouds, we first discretize the system (4.5) and denote it by

$$(6.2) \qquad M_1(\boldsymbol{\mu}) \begin{pmatrix} U \\ V \end{pmatrix} = 0.$$

Similarly, the system (4.6) is discretized and denoted by

$$(6.3) \qquad M_2(\boldsymbol{\mu}) \begin{pmatrix} U \\ V \end{pmatrix} = 0,$$

where

$$(6.4) \qquad M_2(\boldsymbol{\mu}) = \begin{pmatrix} M_3(\boldsymbol{\mu}) & O \\ O & M_3(\boldsymbol{\mu}) \end{pmatrix},$$

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 sections 6.2.1 and 6.2.2.

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

$$(6.5) \qquad (M_1(\boldsymbol{\mu}) + \gamma M_2(\boldsymbol{\mu})) \begin{pmatrix} U \\ V \end{pmatrix} = 0.$$

One advantage of our proposed hybrid scheme is that it is theoretically supported. Since both the system (4.5) and the system (4.6) 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 (6.1).

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 (6.2) produces accurate results but it is highly unstable. By including the second term M_2 in (6.5), we significantly stabilize the computations. On the other hand, the system (6.3) is stable but not accurate. The first term M_1 in (6.5) helps to boost 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 (4.5) and the system (4.6) on point clouds are crucial to the hybrid equation (6.5). 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 (4.5) and (4.6). In this part, we present the numerical schemes to, respectively, approximate the first order PDE system (4.5) and the generalized Laplace equation (4.6) on disk-type point clouds.

For the discretization of (4.5), we adopt a numerical scheme which involves the MLS method [27, 28, 5]. For the discretization of the generalized Laplace equation (4.6), 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, Ho, and Lui [5]:

$$(6.6) \qquad w(d) = \begin{cases} 1 & \text{if } d = 0, \\ \frac{1}{K} \exp\left(\frac{-\sqrt{K}d^2}{D^2}\right) & \text{if } d \neq 0, \end{cases}$$

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 (4.5) 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 (4.5) on the point p, a straightforward calculation yields the following equation,

$$(6.7) \qquad \left(\begin{array}{c} q_2^T(p)A_p \\ -q_1^T(p)A_p \end{array} \right) V = A(p) \left(\begin{array}{c} q_1^T(p)A_p \\ q_2^T(p)A_p \end{array} \right) U.$$

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 $i = 1, 2, \ldots, N$, we have

$$(6.8) \qquad \left(\begin{array}{c} (M_1(\boldsymbol{\mu}))_i \\ (M_1(\boldsymbol{\mu}))_{i+N} \end{array} \right) = \left(\begin{array}{cc} \alpha_1(p_i)q_1^T(p_i) + \alpha_2(p_i)q_2^T(p_i) & -q_2^T(p_i) \\ \alpha_2(p_i)q_1^T(p_i) + \alpha_3(p_i)q_2^T(p_i) & q_1^T(p_i) \end{array} \right) \left(\begin{array}{cc} A_{p_i} & O \\ O & A_{p_i} \end{array} \right),$$

where E_i denotes the ith 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 the 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 (6.7) holds for all $p \in P$, which is true if and only if

$$(6.9) \qquad (q_1(p) + iq_2(p))^T A_p(U + iV) = \boldsymbol{\sigma}(p)(q_1(p) - iq_2(p))^T A_p(U + iV)$$

holds for all $p \in P$. Since the discrete diffuse PCBC is well-defined for all points in P, by a 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 (4.6) 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 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 (4.6) is not constant. Consequently, we need to take its derivatives into considerations. Specifically, we have

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

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 ith row of matrix $M_3(\boldsymbol{\mu})$ is given by

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

where

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

$[\alpha_j(P)]$ denotes a vector whose lth element is $\alpha_j(p_l)$, and E_j denotes the jth row of any 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

$$
(6.13) \qquad (M_3(\boldsymbol{\mu}))_{ij} = \begin{cases} \displaystyle\sum_{T=\{i,j,k\}\in\mathcal{T}_i} \frac{1}{|T|} (p_k - p_i)^T A(T)(p_k - p_j) & \text{if } j \neq i, \\[4ex] \displaystyle\sum_{T=\{i,l,k\}\in\mathcal{T}_i} \frac{1}{|T|} (p_k - p_l)^T A(T)(p_k - p_l) & \text{if } j = i, \end{cases}
$$

where

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

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 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:

1. Conformally parameterizing a disk-type point cloud onto a rectangular planar domain, with the following substeps:
    (a) Mapping the point cloud onto the unit disk by solving the Laplace equation.
    (b) Mapping the unit disk onto the unit square by solving the generalized Laplace equation.
    (c) Optimizing the height of the unit square to achieve conformality.
2. Computing a Teichmüller extremal mapping on the rectangular domain with the prescribed landmark constraints.

6.3.1. Mapping the point cloud onto the unit disk by solving the Laplace equation.

In the coming three subsections, we describe the procedure of conformally parameterizing a disk-type point cloud P onto a rectangle $\tilde{R}$. 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

$$(6.15) \qquad\qquad \Delta\phi_0 = 0$$

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 subsections.

In the discrete case, note that it is hard to approximate the Laplace equation on a surface. Therefore, we perform the computation on a plane with the aid of the following strategy. Recall that in quasi-conformal theory, given an arbitrary diffeomorphism $\varphi : \mathcal{M} \to \varphi(\mathcal{M}) \subset \mathbb{C}$, solving for a harmonic map $\phi_0 : \mathcal{M} \to D$ is equivalent to solving the generalized Laplace equation from $\varphi(\mathcal{M})$ to D with input Beltrami coefficient $\mu(\varphi^{-1})$. In practice, a diffeomorphism from $\mathcal{M}$ to a planar domain is still not easy to solve. Therefore, we locally solve this problem at each point p, and set φ near p to be the projection from the neighborhood of p to its fitting plane computed using principal component analysis. Then, for each point, one linear equation is obtained. By combining all points together, one can get a linear system with solution ϕ_0 being an approximation of the desired harmonic map.

6.3.2. Mapping the unit disk onto the unit square by solving the generalized Laplace equation.

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 $\tilde{R}$ has 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 (4.6).

Proposition 6.2. *Let $f = u + iv : D \to \tilde{R}$ be a quasi-conformal mapping associated with the Beltrami coefficient μ, and $\tilde{R}$ has unit width and height h. Suppose that $f_1 = u_1 + iv_1$ is a quasi-conformal mapping from the unit disk D to the unit square R. If u_1 and v_1 satisfy the generalized Laplace equation (4.6) with the coefficient matrix A computed using μ, with the corresponding boundary condition, then $f = u_1 + ihv_1$.*

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

$$\text{(6.16)} \qquad \frac{1}{\det(g)} \partial_i \left(\sqrt{\det(g)} g^{ij} \partial_j u \right) = 0,$$

$$\text{(6.17)} \qquad \frac{1}{\det(g)} \partial_i \left(\sqrt{\det(g)} g^{ij} \partial_j v \right) = 0.$$

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

$$\text{(6.18)} \qquad \nabla \cdot \left(\sqrt{\det(g)} (g^{ij}) \nabla x \right) = 0.$$

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

$$\text{(6.19)} \qquad A = \sqrt{\det(g)} (g^{ij}).$$

More explicitly, we have

$$\text{(6.20)} \qquad A = \frac{1}{1 - |\mu|^2} \begin{pmatrix} 1 + |\mu|^2 - 2Re(\mu) & -2Im(\mu) \\ -2Im(\mu) & 1 + |\mu|^2 + 2Re(\mu) \end{pmatrix}.$$

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. And the generalized Laplace equation is an elliptic PDE, whose solution is unique with the given boundary condition. Hence, $u = u_1$ and $v/h = v_1$. ∎

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.

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 (6.3) 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.

In the discrete case, to approximate the Beltrami coefficient $\mu(\phi_0^{-1})$, the diffuse PCBC in Definition 4.12 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 (6.3).

6.3.3. Optimizing the height of the unit square to achieve conformality. 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 equal to $\mu(\phi_0^{-1})$. To compute such an optimal h, it is natural to consider the following energy minimization problem,

$$(6.21) \qquad h = argmin \int_{\mathbb{D}} |\mu(\phi_2) - \mu(\phi_0^{-1})|^2.$$

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

$$(6.22) \qquad \phi = \phi_2 \circ \phi_0.$$

In the discrete case, 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, it is natural to replace the integral in (6.21) by a summation. Now, it remains to solve the following minimization problem

$$(6.23) \qquad h = argmin \ \|\tilde{\sigma} - \tilde{\mu}\|^2,$$

where $\tilde{\sigma}$ and $\tilde{\mu}$ 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 and the computation for h becomes straightforward.

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

6.3.4. 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 Teichmüller mappings 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 Teichmüller mapping 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 $\tilde{\boldsymbol{\mu}}_n$ of a given point cloud mapping $\mathbf{f}_n$ using the definition of PCBC in Definition 4.12.

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 Teichmüller mapping. 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 $\tilde{\boldsymbol{\mu}}_n$ at the points where the norms are less than 1. In other words, we take

$$(6.24) \qquad k_n = \begin{cases} \frac{\sum_{p \in S_n} |\tilde{\boldsymbol{\mu}}_n(p)|}{|S_n|} & \text{if } |S_n| > 0, \\ 0 & \text{if } |S_n| = 0, \end{cases}$$

where

$$(6.25) \qquad S_n = \{p \in P : |\tilde{\boldsymbol{\mu}}_n(p)| < 1\}.$$

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

$$(6.26) \qquad \boldsymbol{\nu}_n(p_i) = \begin{cases} \dfrac{\tilde{\boldsymbol{\mu}}_n(p_i)}{|\tilde{\boldsymbol{\mu}}_n(p_i)|} & \text{if } \tilde{\boldsymbol{\mu}}_n(p_i) \neq 0, \\ 1 & \text{otherwise.} \end{cases}$$

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

$$(6.27) \qquad \boldsymbol{\tau}_n(p_i) = \begin{cases} \dfrac{L_i \boldsymbol{\nu}_n}{|L_i \boldsymbol{\nu}_n|} & \text{if } L_i \boldsymbol{\nu}_n \neq 0, \\ \boldsymbol{\nu}_n(p_i) & \text{otherwise.} \end{cases}$$

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 $k_n = \begin{cases} \frac{\sum_{p \in S_n} |\tilde{\boldsymbol{\mu}}_n(p)|}{|S_n|} & \text{if } |S_n| > 0, \\ 0 & \text{if } |S_n| = 0, \end{cases}$ 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

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

      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 (6.5) 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 $\mathbf{f} = \mathbf{f}_n$.

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

$$(6.28) \qquad\qquad\qquad\qquad \boldsymbol{\sigma}_n = k_n \boldsymbol{\tau}_n.$$

In the last step, a new mapping $\mathbf{f}_{n+1}$ is computed by solving the hybrid equation (6.5) 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 6.3. 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* $\lim_n \gamma_n = 0$, *where* γ_n *denotes the relative weight parameter for the hybrid equation* (6.5) *in the nth iteration. 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 5.3. By definition, for any point $x \in P$,

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

By the MLS error estimate in (4.20), for $j = 1, 2$,

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

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 $j = 1, 2$,

$$
(6.31) \quad \lim_n \left| D_j\left[\begin{pmatrix} U_n \\ V_n \end{pmatrix} - \begin{pmatrix} U \\ V \end{pmatrix} \right] \right| = 0.
$$

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 (6.5) with input PCBC $\boldsymbol{\sigma}_n$. Let

$$
(6.32) \quad E_n = -\gamma_n M_2(\boldsymbol{\sigma}_n) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix},
$$

then

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

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

$$
(6.34) \quad
\begin{aligned}
&D_1(x) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} \\
&= q_1^T(x)A_x U_{n+1} + iq_2^T(x)A_x U_{n+1} + iq_1^T(x)A_x V_{n+1} - q_2^T(x)A_x V_{n+1} \\
&= q_1^T(x)A_x U_{n+1} + iq_2^T(x)A_x U_{n+1} - i(\alpha_2(\boldsymbol{\sigma}_n|_x)q_1^T(x) + \alpha_3(\boldsymbol{\sigma}_n|_x)q_2^T(x))A_x U_{n+1} \\
&\quad - (\alpha_1(\boldsymbol{\sigma}_n|_x)q_1^T(x) + \alpha_2(\boldsymbol{\sigma}_n|_x)q_2^T(x))A_x U_{n+1} + i(E_n)_{j+N} + (E_n)_j \\
&= \left(2\frac{\boldsymbol{\sigma}_n(x) - |\boldsymbol{\sigma}_n(x)|^2}{1 - |\boldsymbol{\sigma}_n(x)|^2}q_1(x) - 2i\frac{\boldsymbol{\sigma}_n(x) + |\boldsymbol{\sigma}_n(x)|^2}{1 - |\boldsymbol{\sigma}_n(x)|^2}q_2(x) \right)^T A_x U_{n+1} \\
&\quad + i(E_n)_{j+N} + (E_n)_j.
\end{aligned}
$$

Similarly, the following equation also holds:

$$(6.35) \quad \begin{aligned} &D_2(x) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} \\ &= \left(2\frac{1 - \overline{\boldsymbol{\sigma}_n(x)}}{1 - |\boldsymbol{\sigma}_n(x)|^2} q_1(x) - 2i\frac{1 + \overline{\boldsymbol{\sigma}_n(x)}}{1 - |\boldsymbol{\sigma}_n(x)|^2} q_2(x) \right)^T A_x U_{n+1} + i(E_n)_{j+N} - (E_n)_j. \end{aligned}$$

Then,

$$(6.36) \quad \boldsymbol{\sigma}_n(x) = \frac{D_1(x) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} - i(E_n)_{j+N} - (E_n)_j}{D_2(x) \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} - i(E_n)_{j+N} + (E_n)_j}.$$

Since the point cloud P is fixed and $|\boldsymbol{\sigma}_n(x)| \le k_n < 1$ 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 (6.31) 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 (6.36) 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 $L_i\tilde{\boldsymbol{\mu}} \ne 0$ and $k \ne 0$. 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 $\tilde{\boldsymbol{\mu}}_2(p_i) \ne 0$. Then, when n is large enough,

$$(6.37) \quad \frac{\tilde{\boldsymbol{\rho}}_1(p_i)}{\tilde{\boldsymbol{\rho}}_2(p_i)} = \lim_n \frac{\boldsymbol{\sigma}_{n,1}(p_i)}{\boldsymbol{\sigma}_{n,2}(p_i)} = \lim_n \frac{L_i(\tilde{\boldsymbol{\mu}}_{n,1}/|\tilde{\boldsymbol{\mu}}_n|)}{L_i(\tilde{\boldsymbol{\mu}}_{n,2}/|\tilde{\boldsymbol{\mu}}_n|)} = \frac{L_i(\tilde{\boldsymbol{\rho}}_1/k)}{L_i(\tilde{\boldsymbol{\rho}}_2/k)} = \frac{L_i\tilde{\boldsymbol{\rho}}_1}{L_i\tilde{\boldsymbol{\rho}}_2}.$$

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 $\tilde{\boldsymbol{\mu}}_n$, $\boldsymbol{\sigma}_n$ 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 5.3 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. A current limitation of the above proposition is the key assumption on the convergence of Algorithm 2. One of our future papers will be proving the convergence of Algorithm 2.

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 the 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 dissimilarity metric based on the Teichmüller parameterizations for classifying different point clouds with prescribed landmark constraints.

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

$$(6.38) \qquad\qquad\qquad\qquad p_i \leftrightarrow q_i$$

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

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,

$$(6.39) \qquad\qquad\qquad\qquad h(g_1(p_i)) = g_2(q_i)$$

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

$$(6.40) \qquad\qquad\qquad\qquad f = g_2^{-1} \circ h \circ g_1.$$

Obviously, f satisfies

$$(6.41) \qquad\qquad\qquad\qquad f(p_i) = q_i.$$

Moreover, f is a Teichmüller mapping. In other words, the Beltrami coefficient associated with f has 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 has constant norm. Also, since g_1, g_2 are conformal mappings, the Beltrami coefficient μ_{g_1}, μ_{g_2} is zero. By Theorem 4.3,

$$(6.42) \qquad \mu_f = \mu_{g_2^{-1} \circ h \circ g_1} = (\mu_h \circ g_1)\frac{\overline{(\partial_z g_1)}}{\partial_z g_1} = k\frac{\overline{(\partial_z g_1)^2 (\varphi \circ g_1)}}{|(\partial_z g_1)^2 (\varphi \circ g_1)|},$$

where $k = |\mu_h|$ is a constant and φ is holomorphic. Moreover, since g_1 is conformal, we have $(\partial_z g_1)^2 (\varphi \circ g_1)$ 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 Algorithms 1 and 2. The entire registration algorithm is summarized in Algorithm 3.

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

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 $\mathbf{f} = \mathbf{g}_2^{-1} \circ \mathbf{h} \circ \mathbf{g}_1$.

Definition 6.4 (Teichmüller metric). *For every i, let M_i be a Riemann surface with feature landmarks $\{p_{ik}\}_{k=1}^n$. The* Teichmüller metric *between (f_i, M_i) and (f_j, M_j) is defined as*

$$(6.43) \qquad d_T((f_i, M_i), (f_j, M_j)) = \inf_\varphi \frac{1}{2} \log K(\varphi),$$

where $\varphi : M_i \to M_j$ varies over all quasi-conformal mappings with $\{p_{ik}\}_{k=1}^n$ corresponding 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

$$(6.44) \qquad K(\varphi) = K(h).$$

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

By Theorem 4.7, with a suitable boundary condition, the Teichmüller mapping between two unit disks is extremal and unique. A similar result also holds for the Teichmüller mapping between two rectangular domains. Therefore, the infimum in (6.43) is achieved by the unique Teichmüller mapping between $\tilde{R}_i$ and $\tilde{R}_j$. To be more specific, we have

$$(6.45) \qquad d_T((f_i, M_i), (f_j, M_j)) = \frac{1}{2} \log K(h),$$

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 the following method for building up a dissimilarity metric of feature-endowed point clouds.

Let $\{P_t\}$ be a collection of feature-endowed point clouds. To perform an accurate classification on $\{P_t\}$, we take every pair of point cloud surface P_i, P_j and compute the

---

Algorithm 4. Building a distance matrix using the Teichmüller metric.

Input: A set of point clouds $\{P_t\}_{t=1}^{M}$ with landmark correspondences
Output: The distance matrix $D = (d_{ij})$, $i, j = 1, 2, \ldots, M$

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

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

3 Update D by $(D + D^T)/2$;

---

landmark-matching Teichmüller mappings $\mathbf{f}_{ij} : P_i \to P_j$ for all i, j. Then, we evaluate the associated discrete diffuse PCBCs $\boldsymbol{\mu}_{ij}$ and denote the Teichmüller distance of P_i and P_j by

$$(6.46) \qquad d_{ij} = \frac{1}{2} \log \frac{1 + \|\boldsymbol{\mu}_{ij}\|_{\infty}}{1 - \|\boldsymbol{\mu}_{ij}\|_{\infty}}.$$

The value d_{ij} records the difference between the point clouds P_i and P_j. Specifically, if P_i and P_j have similar shapes, then d_{ij} is small. By analyzing the distance matrix (d_{ij}) formed, we can effectively classify different feature-endowed point clouds. For instance, the multidimensional scaling (MDS) method can be applied for the classification here.

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 $|\mu|$. It is also noteworthy that, in practice, given two point clouds P_i and P_j, the Teichmüller distance d_{ij} (induced by the Teichmüller mapping $f_{ij} : P_i \to P_j$) may be slightly different from d_{ji} (induced by the Teichmüller mapping $f_{ji} : P_j \to P_i$) because of the numerical errors and the stopping criterion of the algorithm. Therefore, for symmetry, we replace the distance matrix D by $(D + D^T)/2$. Our algorithm for the dissimilarity metric is summarized in Algorithm 4. Experimental results are provided in section 7 to illustrate the effectiveness of our proposed method.

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] and the TF3DM repository [54]. The Laboratory for Computational Longitudinal Neuroimaging shape database [53], the human face dataset [52], the Spacetime Face Data [55], and some additional facial point clouds sampled using Kinect are used in the first shape analysis experiment. The database in [56] is used in the second 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.
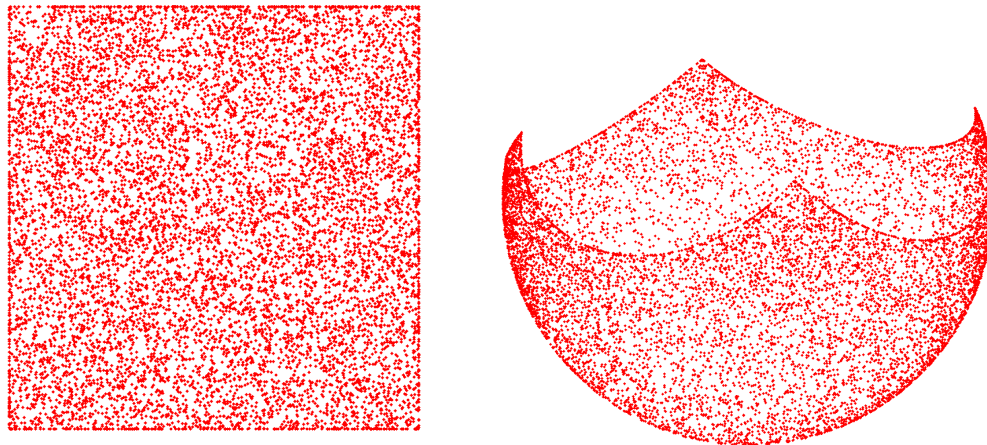
Figure 3. *An example (the stereographic projection) used in evaluating different numerical schemes for computing conformal mappings on point clouds. The conformal mapping is given by $f(x,y) = \left(\frac{2x}{x^2+y^2+1}, \frac{2y}{x^2+y^2+1}, \frac{x^2+y^2-1}{x^2+y^2+1} \right)$.*

We first compare our proposed scheme with the local mesh method [24] and the MLS 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 planar point cloud to a 3D point cloud using a conformal mapping with an explicit formula. 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. An example of such a conformal mapping with an explicit formula is given in Figure 3. Table 3 lists the statistics of several experiments with this conformal map. 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 experiment, we transform a randomly generated point cloud using a quasi-conformal mapping with an explicit formula. An example is given in Figure 4. 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 4 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.

Table 3

The performances of three numerical schemes for computing conformal maps on point clouds. The stereographic projection with a randomly generated point cloud is considered in each experiment.

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

Example 4 (noisy)

Method	Maximum position error	Average 1-norm error	Average 2-norm error
Local mesh [24]	0.050	0.0150	0.000202
MLS with special weight [27, 28]	0.379	0.0881	0.009463
Our proposed method	0.026	0.0083	0.000063

Example 5 (noisy)

Method	Maximum position error	Average 1-norm error	Average 2-norm error
Local mesh [24]	0.090	0.0128	0.000238
MLS with special weight [27, 28]	0.501	0.0574	0.004941
Our proposed method	0.049	0.0097	0.000105

Note that there is a balancing parameter γ in the hybrid equation (6.5) of our TEMPO algorithm. In this section, we first compare Teichmüller parameterization results with different choices of γ. Two categories of the parameter γ are considered in running the TEMPO algorithm:

(i) Decreasing values: γ_n is gradually decreased during the iterations.

(ii) Constant values: γ is set to be a constant throughout the iterations.

In the experiment, we consider computing the Teichmüller parameterization of a real 3D point cloud. Since we do not have any explicit formula as the ground truth solution, we evaluate the performance by computing the variance of $|\boldsymbol{\mu}|$. It is desirable for the Teichmüller parameterization to have a negligible variance of $|\boldsymbol{\mu}|$.

The parameterization results with different input γ are recorded in Table 5. For fairness, we perform the same number of iterations in Algorithm 2. It is noteworthy that the cases $\gamma = 0$ or ∞, respectively, correspond to solely solving (4.5) or (4.6).
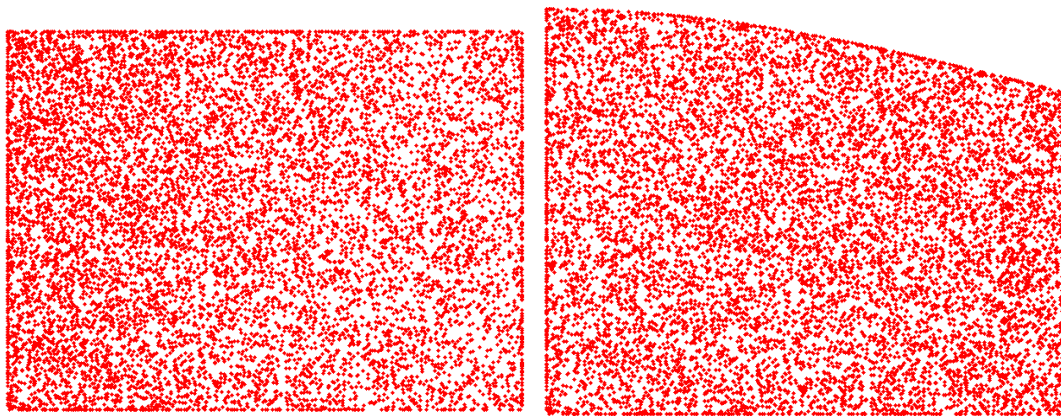
Figure 4. *An example used in evaluating different schemes for computing quasi-conformal mappings on point clouds. The quasi-conformal mapping is given by* $f(x, y) = \left(\log(x+1), \arcsin\left(\frac{y}{2+(\log(x+1))^2} \right) \right).$

It can be observed that if $\gamma = 0$, the algorithm fails. On the other hand, the result associated with $\gamma = \infty$ is not as accurate as those associated with nonzero finite γ. Consequently, our hybrid scheme outperforms the two existing methods.

In addition, the results associated with different nonzero finite values of γ are highly similar, regardless of whether γ is decreasing or constant. This observation suggests that the assumption $\lim_n \gamma_n = 0$ in Proposition 6.3 can be relaxed in practice. Therefore, for simplicity, we can take a constant γ in running the TEMPO algorithm.

In the following experiments, the balancing parameter γ in the hybrid equation (6.5) is set to be $\gamma \equiv 0.5$. The stopping criterion is set to be $\epsilon = 10^{-6}$.

Figure 5 shows the landmark-constrained Teichmüller parameterization result of a spiral 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 Figures 6 and 7. 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.

Noticing that real point clouds are usually with certain undesirable noises, we also test the performance of our TEMPO algorithm on noisy point clouds. Figure 8 shows a noisy facial point cloud with 5% uniformly distributed random noise and the Teichmüller parameterization result. The histogram of the resulting Beltrami coefficient indicates that the resulting map is a Teichmüller extremal map. This demonstrates the effectiveness of our TEMPO algorithm on noisy point cloud data.

Extensive experiments with 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 6. Our proposed method is highly efficient. The computations of the landmark-matching Teichmüller parameterizations complete within

Table 4

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

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

Example 4 (noisy)

Method	Maximum position error	Average 1-norm error	Average 2-norm error
Local mesh [24]	0.0181	0.00685	0.00004237
MLS with special weight [27, 28]	0.0565	0.01594	0.00034875
Our proposed method	0.0177	0.00652	0.00003953

Example 5 (noisy)

Method	Maximum position error	Average 1-norm error	Average 2-norm error
Local mesh [24]	0.0170	0.00769	0.00005009
MLS with special weight [27, 28]	0.0429	0.01948	0.00034534
Our proposed method	0.0170	0.00762	0.00004964

Table 5

The results obtained by different choices of γ in Algorithm 2. The row of γ refers to the initial values.

Result with decreasing γ

γ	0	1	50	100	500	1000	∞
Var($|\boldsymbol{\mu}|$)	NaN	0.0018	0.0024	0.0020	0.0105	0.0085	0.0050

Result with constant γ

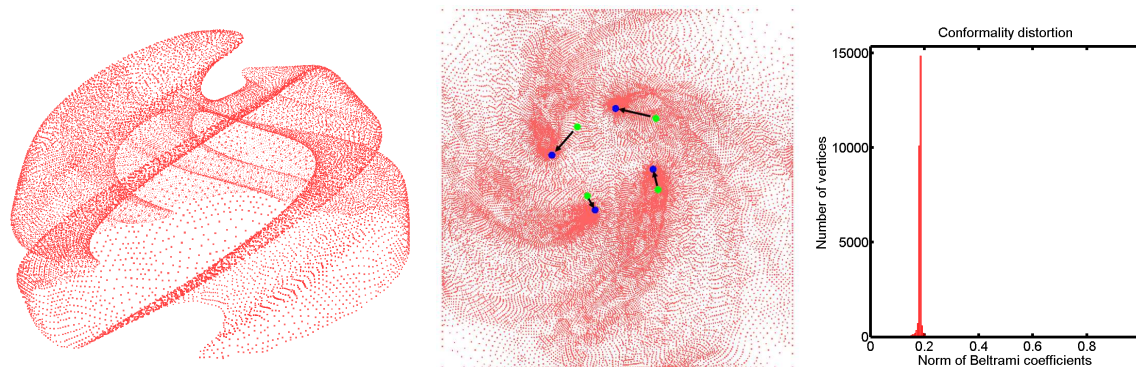| γ	0	1	50	100	500	1000	∞
Var($|\boldsymbol{\mu}|$)	NaN	0.0018	0.0022	0.0017	0.0031	0.0022	0.0050

Figure 5. *Teichmüller parameterization of a spiral 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.*
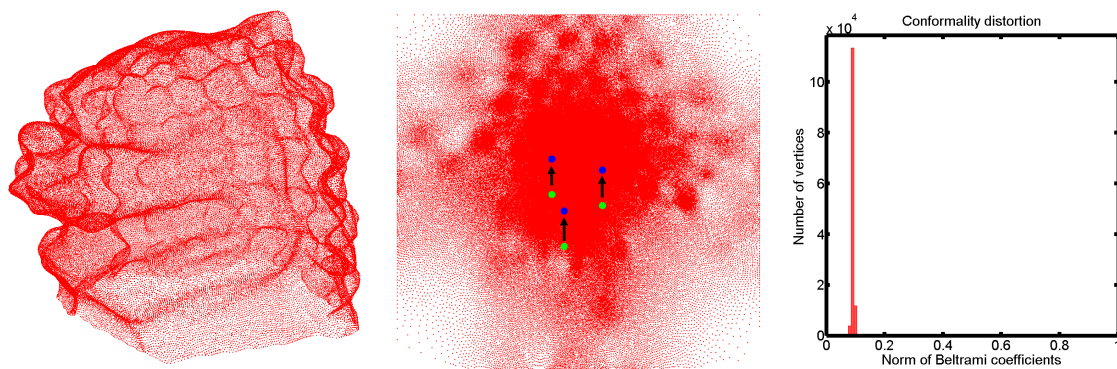
Figure 6. *Teichmüller parameterization of a lion point cloud. Left: The input point cloud. Middle: The parameterization result. The landmark constraints are represented by the green and blue points. The green points represent the original locations and the blue points are the target locations. Right: The histogram of the norm of the Beltrami coefficients.*

1 minute on average for point clouds with moderate 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. It can also be observed in Table 6 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 registration between point clouds with
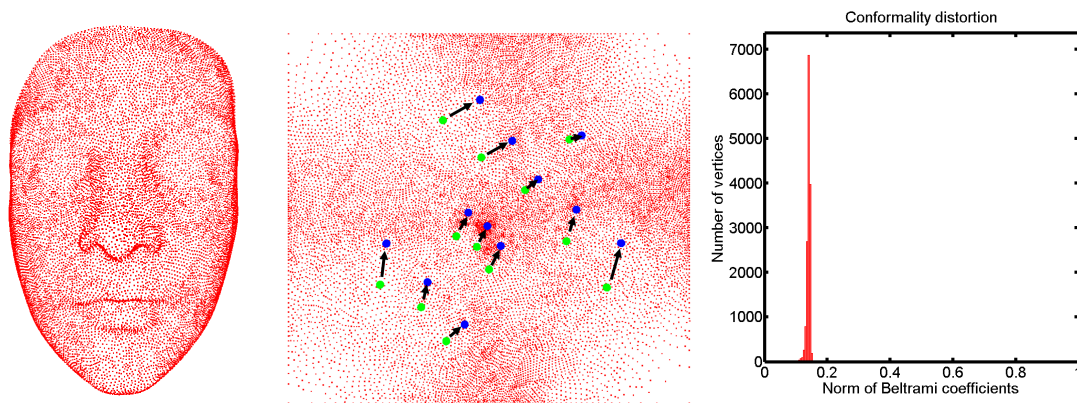
Figure 7. *Teichmüller parameterization of Lucy point cloud. Left: The input point cloud. Middle: The parameterization result. The landmark constraints are represented by the green and blue points. The green points represent the original locations and the blue points are the target locations. Right: The histogram of the norm of the Beltrami coefficients.*
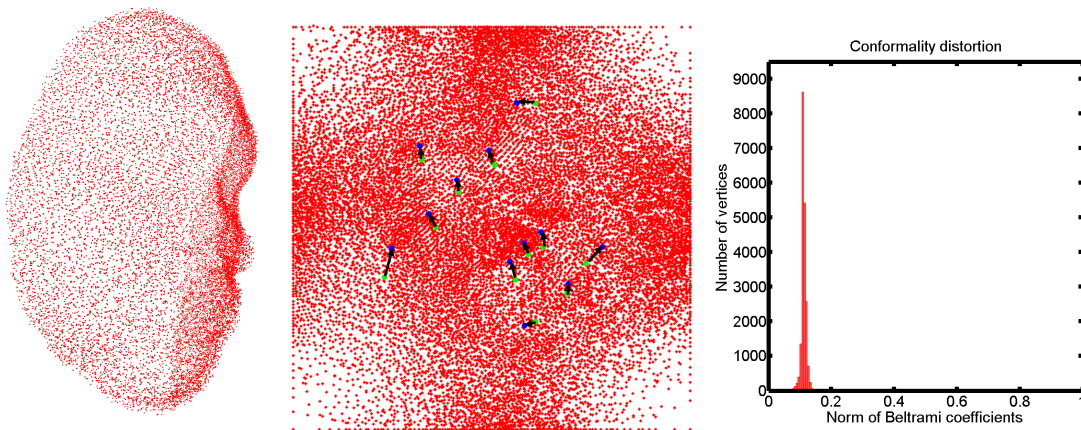

Figure 8. *Teichmüller parameterization of a noisy facial point cloud. Left: The input noisy point cloud. Middle: The parameterization result. The landmark constraints are represented by the green and blue points. The green points represent the original locations and the blue points are the target locations. Right: The histogram of the norm of the Beltrami coefficients.*

prescribed landmark constraints can be efficiently computed as described in Algorithm 3. Figure 9 shows the registration of two human face point 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.

The Teichmüller metric induced by the landmark matching Teichmüller parameterization serves as an effective dissimilarity metric for shape analysis of point clouds. To demonstrate the effectiveness of the Teichmüller metric, we consider a facial point cloud classification problem. Suppose we are given a set of facial point clouds. It is conceivable that even with

Table 6
The performance of our landmark-constrained Teichmüller parameterization algorithm for various point clouds.

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
Spiral	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
Noisy face 1	20184	0.9172	0.1111	0.0140
Noisy face 2	19743	0.5191	0.1077	0.0130
Noisy face 3	20194	0.4905	0.0787	0.0157

different facial expressions, point clouds sampled from the same human should be similar in shape. We aim to correctly classify the point clouds into several groups, where each of the groups represents one human.

In our experiments, each subject has multiple facial expressions. Landmark constraints at the most prominent parts of the faces, such as the eyes, nose, and mouth of each point cloud are manually labeled to ensure the accuracy of the classification. We compare our Teichmüller metric with another dissimilarity metric computed by a feature-endowed point cloud mapping algorithm in [20]. For a fair comparison, the automatically detected landmarks in [20] are replaced by our manually labeled landmarks. In the experiments, we first compute our proposed distance matrix using Algorithm 4. On the other hand, we compute another distance matrix using [20]. To assess the classification performance, we apply the MDS method on the two distance matrices. Two experiments are presented below.

In the first experiment, we are interested in classifying facial point clouds with prominent facial expressions. 16 facial point clouds with 4 specific expressions (neutral, happy, sad, angry) are adapted from [53, 52, 55] or sampled by Kinect. Figure 10 shows the dataset. 12 landmark constraints are manually labeled on each point cloud for the computation. The MDS results of our distance matrix and the distance matrix in [20] are shown in Figure 11. It can be observed that distinct subjects are effectively clustered using our distance matrix. Even with highly different facial expressions, the facial point clouds sampled from the same subject can be grouped. On the contrary, the result based on the distance matrix in [20] cannot identify distinct subjects. This implies that the Teichmüller metric leads to a better classification result when compared with the distance matrix in [20]. For a more detailed comparison, the leave-one-out cross-validation (LOOCV) is applied to evaluate the classification accuracies based on our distance matrix and the distance matrix computed in [20]. Our distance matrix results in a classification accuracy of 94%, while the distance matrix in [20] results in a classification accuracy of 50%. The comparison demonstrates the effectiveness of our TEMPO method in shape analysis of point clouds.
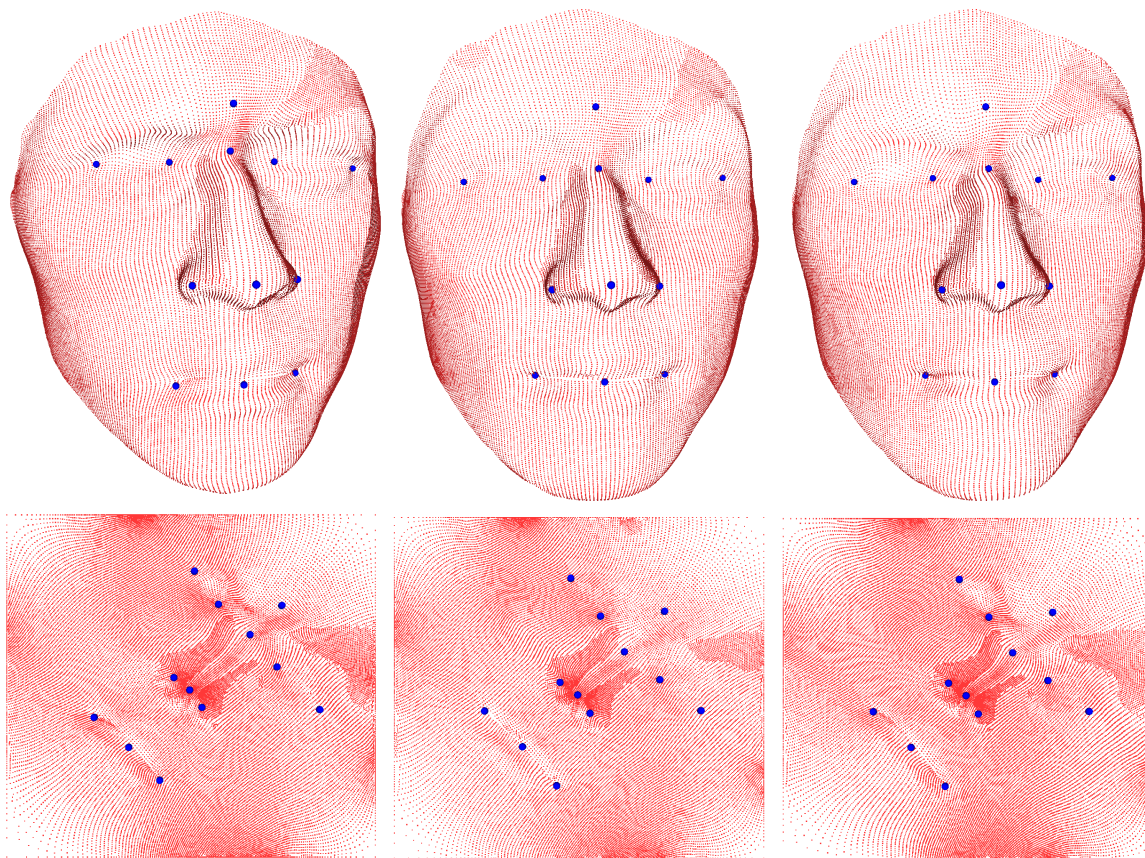
Figure 9. *Teichmüller registration between two human face point clouds with prescribed landmark constraints. Top left: The source point cloud. Top middle: The target point cloud. Top right: The registration result by our algorithm. Bottom: The conformal parameterizations of the source and target point clouds, and the landmark-constrained Teichmüller parameterization. The blue points refer to the facial landmark constraints.*
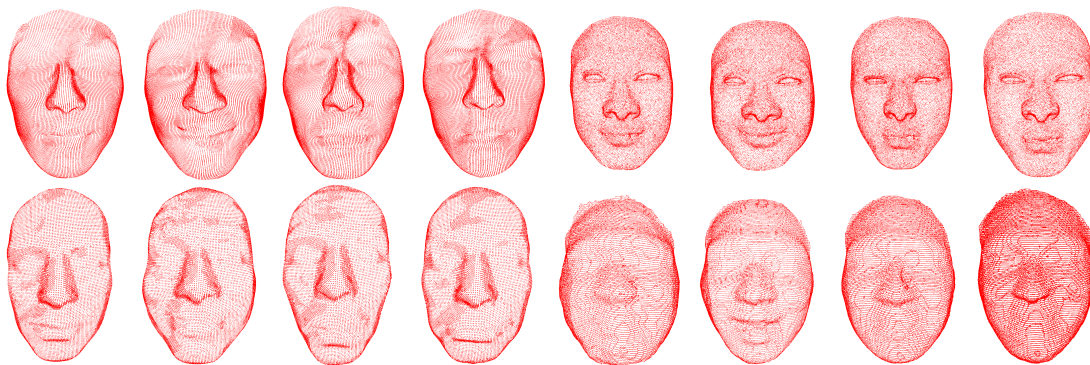

Figure 10. *The dataset of human face point clouds with different facial expressions used in our first experiment.*
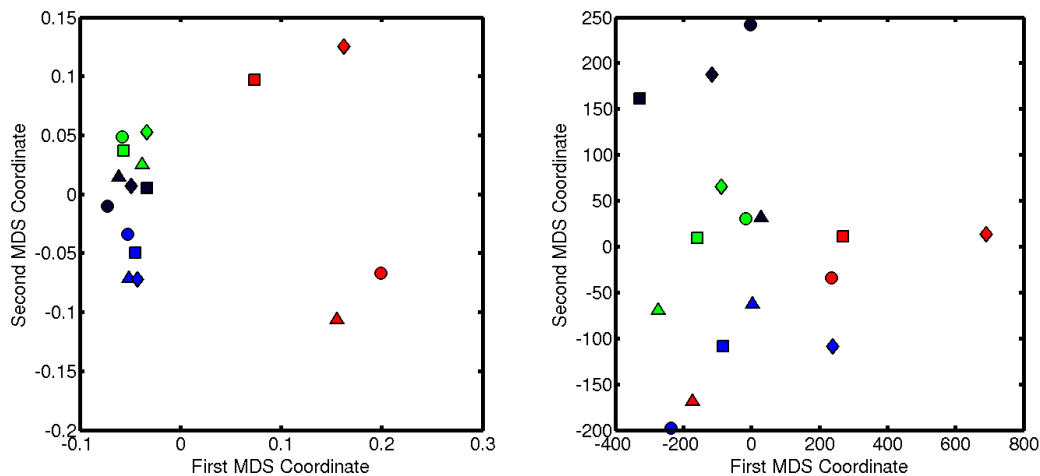
Figure 11. *The MDS results for our distance matrix (left) and the distance matrix in* [20] *(right). Each color represents one human, and each shape represents a type of facial expressions (circle: neutral; square: happy; triangle: sad; diamond: angry).*

In the second experiment, we consider a larger dataset. More specifically, we adapt the 3D human face database [56], which is the database used in [20]. Each subject in the database has 3 random facial expressions. Figure 12 shows a partial set of the facial point clouds in [56]. 10 landmark constraints on each point cloud are labeled for the computation. Note that in our experiment, we do not perform any triangulation or smoothing procedure on the raw point cloud data. Our distance matrix results in an LOOCV accuracy of 79%, while the distance matrix in [20] results in an LOOCV accuracy of 50%. The results again reflect the effectiveness of our proposed dissimilarity metric.

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. The consistency between the PCT-maps and the continuous Teichmüller mappings 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. First, we have introduced a hybrid quasi-conformal reconstruction scheme for computing quasi-conformal mappings on point clouds. Second, an improved method has been proposed for approximating differential operators on point clouds with disk topology. The new approximation method produces more accurate results. Third, 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. Fourth, 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.

Figure 12. *A partial set of the facial point clouds used in our second experiment, adapted from the* 3D *human face database* [56].

REFERENCES

[1] P. Breitkopf, H. Naceur, A. Rassineux, and P. Villon, *Moving least squares response surface approximation: Formulation and metal forming applications*, Comput. Struct., 83 (2005), pp. 1411–1428.

[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 J. Imaging Sci., 8 (2015), pp. 67–94.

[3] P. T. Choi and L. M. Lui, *Fast disk conformal parameterization of simply-connected open surfaces*, J. Sci. Comput., 65 (2015), pp. 1065–1090.

[4] G. P.-T. Choi and L. M. Lui, *A Linear Formulation for Disk Conformal Parameterization of Simply-Connected Open Surfaces*, preprint, arXiv:1508.00396, 2015.

[5] G. P.-T. Choi, K. T. Ho, and L. M. Lui, *Spherical conformal parameterization of genus-0 point clouds for meshing*, SIAM J. Imaging Sci., 9 (2016), pp. 1582–1618.

[6] M. Desbrun, M. Meyer, and P. Alliez, *Intrinsic parameterizations of surface meshes*, Comput. Graph. Forum, 21 (2002), pp. 209–218.

[7] M. Floater, *Parameterization and smooth approximation of surface triangulations*, Comput. Aided Geom. Design, 14 (1997), pp. 231–250.

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

[9] M. S. FLOATER AND M. REIMERS, *Meshless parameterization and surface reconstruction*, Comput. Aided Geom. Design, 18 (2001), pp. 77–92.

[10] M. FLOATER AND K. HORMANN, *Parameterization of triangulations and unorganized points*, in Tutorials on Multiresolution in Geometric Modeling, Springer, Berlin, 2002, pp. 287–316.

[11] M. FLOATER, *Mean value coordinates*, Comput. Aided Geom. Design, 20 (2003), pp. 19–27.

[12] M. FLOATER AND K. HORMANN, *Surface parameterization: A tutorial and survey*, in Advances in Multiresolution for Geometric Modeling, Springer, Berlin, 2005, pp. 157–186.

[13] F. GARDINER AND N. LAKIC *Quasiconformal Teichmüller Theory*, American Mathematics Society, Providence, RI, 2000.

[14] X. GU AND S. T. YAU, *Computing conformal structures of surfaces*, Commun. Inf. Syst., 2 (2002), pp. 121–146.

[15] X. GU AND S. T. YAU, *Global conformal surface parameterization*, in Eurographics Symposium on Geometry Processing, ACM, New York, 2003, pp. 127–137.

[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 Trans. Med. Imaging, 23 (2004), pp. 949–958.

[17] X. GUO, X. LI, Y. BAO, X. GU, AND H. QIN, *Meshless thin-shell simulation based on global conformal parameterization*, IEEE Trans. Vis. Comput. Graph., 12 (2006), pp. 375–385.

[18] K. HORMANN AND G. GREINER, *MIPS: An efficient global parametrization method*, in Curve and Surface Design, Vanderbilt University Press, Nashville, TN, 2000, pp. 153–162.

[19] K. HORMANN, B. LÉVY, AND A. SHEFFER, *Mesh parameterization: Theory and practice*, SIGGRAPH 2007 ACM, New York, 2007, 1.

[20] M. O. İRFANOĞLU, B. GÖKBERK, AND L. AKARUN, 3*D shape-based face recognition using automatically registered facial surfaces*, in Proceedings of the 17th International Conference on Pattern Recognition, IEEE Computer Society, Los Alamitos, CA, 2004, pp. 183–186.

[21] M. JIN, Y. WANG, X. GU, AND S. T. YAU, *Optimal global conformal surface parameterization for visualization*, Commun. Inf. Syst., 4 (2005), pp. 117–134.

[22] M. JIN, J. KIM, F. LUO, AND X. GU, *Discrete surface Ricci flow*, IEEE Trans. Vis. Comput. Graph., 14 (2008), pp. 1030–1043.

[23] L. KHAREVYCH, B. SPRINGBORN, AND P. SCHRÖDER, *Discrete conformal mappings via circle patterns*, ACM Trans. Graph., 25 (2006), pp. 412–438.

[24] R. LAI, J. LIANG, AND H. ZHAO, *A local mesh method for solving PDEs on point clouds*, Inverse Probl. Imaging, 7 (2013), pp. 737–755.

[25] D. LEVIN, *The approximation power of moving least-squares*, Math. Comput., 67 (1998), pp. 1517–1531.

[26] B. LÉVY, S. PETITJEAN, N. RAY, AND J. MAILLOT, *Least squares conformal maps for automatic texture atlas generation*, in ACM SIGGRAPH 2002, ACM, New York, 2002, pp. 362–371.

[27] J. LIANG, R. LAI, A. WONG, AND H. ZHAO, *Geometric understanding of point clouds using Laplace-Beltrami operator*, in IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Piscataway, NJ, 2012, pp. 214–221.

[28] J. LIANG AND H. ZHAO, *Solving partial differential equations on point clouds*, SIAM J. Sci. Comput., 35 (2013), pp. A1461–A1486.

[29] Y. LIPMAN, *Bounded distortion mapping spaces for triangular meshes*, ACM Trans. Graph., 31 (2012), 108.

[30] W.K. LIU, S. LI, AND T. BELYTSCHKO, *Moving least-square reproducing kernel methods (I) methodology and convergence*, Comput. Methods Appl. Mech. Engrg., 143 (1997), pp. 113–154.

[31] L. M. LUI, K. C. LAM, T. W. WONG, AND X. GU, *Texture map and video compression using Beltrami representation*, SIAM J. Imaging Sci., 6 (2013), pp. 1880–1902.

[32] L. M. LUI, K. C. LAM, S.-T. YAU, AND X. GU, *Teichmüller mapping (T-map) and its applications to landmark matching registration*, SIAM J. Imaging Sci., 7 (2014), pp. 391–426.

[33] L. M. LUI, X. GU, AND S.-T. YAU, *Convergence of an iterative algorithm for Teichmüller maps via harmonic energy optimization*, Math. Comput., 84 (2015), pp. 2823–2842.

[34] F. LUO, *Combinatorial Yamabe flow on surfaces*, Comm. Contemp. Math., 6 (2004), pp. 765–780.

[35] T. W. MENG AND L. M. LUI, *The Theory of Computational Quasi-Conformal Geometry on Point Clouds*, preprint, arXiv:1510.05104v1, 2015.

[36] D. Mirzaei, R. Schaback, and M. Dehghan, *On generalized moving least squares and diffuse derivatives*, IMA J. Numer. Anal., 32 (2012), pp. 983–1000.

[37] D. Mirzaei, *Analysis of moving least squares approximation revisited*, J. Comput. Appl. Math., 282 (2015), pp. 237–250.

[38] P. Mullen, Y. Tong, P. Alliez, and M. Desbrun, *Spectral conformal parameterization*, Comput. Graph. Forum, 27 (2008), pp. 1487–1494.

[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, Darmstadt, Germany, 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*, in Proceedings of 6th International Meshing Roundtable, World Scientific, Singapore, 2000, pp. 161–172.

[42] A. Sheffer, B. Lévy, M. Mogilnitsky, and A. Bogomyakov, *ABF++: Fast and robust angle based flattening*, ACM Trans. Graph., 24 (2005), pp. 311–330.

[43] A. Sheffer, E. Praun, and K. Rose, *Mesh parameterization methods and their applications*, Found. Trends Comput. Graph. Vision, 2 (2006), pp. 105–171.

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

[45] H. Wendland, *Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree*, Adv. Comput. Math., 4 (1995), pp. 389–396.

[46] H. Wendland, *Local polynomial reproduction and moving least squares approximation*, IMA J. Numer. Anal., 21 (2001), pp. 285–300.

[47] H. Wendland, *Scattered Data Approximation*, Cambridge Monogr. Appl. Comput. Math. 17, Cambridge University Press, Cambridge, 2004.

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

[49] L. Zhang, L. Liu, C. Gotsman, and H. Huang, *Mesh reconstruction by meshless denoising and parameterization*, Comput. Graph., 34 (2010), pp. 198–208.

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

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

[52] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, *Calculus of non-rigid surfaces for geometry and texture manipulation*, IEEE Trans. Vis. Comput. Graph., 13 (2007), pp. 902–913.

[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 Trans. Graph., 23 (2004), pp. 548–558.

[56] C. Beumier and M. Acheroy, *Automatic 3D face authentication*, Image Vision Comput., 18 (2000), pp. 315–321.