DOI: 10.1111/cgf.13315

COMPUTER GRAPHICS *forum*
Volume 00 (2017), number 0 pp. 1–13

Laplace–Beltrami Operator on Point Clouds Based on Anisotropic Voronoi Diagram

Hongxing Qin[1,2], Yi Chen[1,2], Yunhai Wang[3,*], Xiaoyang Hong[1,2], Kangkang Yin[4] and Hui Huang[5]

[1]Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts & Telecommunications, Chongqing, China
redstarq@gmail.com, chenyiwowo@163.com, hxy900605@qq.com
[2]College of Computer Science and Technology, Chongqing University of Posts & Telecommunications, Chongqing, China
[3]Shandong University, Jinan, China
cloudseawang@gmail.com
[4]Simon Fraser University, Canada
kkyin@sfu.ca
[5]Shenzhen University, Shenzhen, China
huihuang@szu.edu.cn

Abstract

The symmetrizable and converged Laplace–Beltrami operator ($\Delta_{\mathcal{M}}$) is an indispensable tool for spectral geometrical analysis of point clouds. The $\Delta_{\mathcal{M}}$, introduced by Liu et al. [LPG12] is guaranteed to be symmetrizable, but its convergence degrades when it is applied to models with sharp features. In this paper, we propose a novel $\Delta_{\mathcal{M}}$, which is not only symmetrizable but also can handle the point-sampled surface containing significant sharp features. By constructing the anisotropic Voronoi diagram in the local tangential space, the $\Delta_{\mathcal{M}}$ can be well constructed for any given point. To compute the area of anisotropic Voronoi cell, we introduce an efficient approximation by projecting the cell to the local tangent plane and have proved its convergence. We present numerical experiments that clearly demonstrate the robustness and efficiency of the proposed $\Delta_{\mathcal{M}}$ for point clouds that may contain noise, outliers, and non-uniformities in thickness and spacing. Moreover, we can show that its spectrum is more accurate than the ones from existing $\Delta_{\mathcal{M}}$ for scan points or surfaces with sharp features.

Keywords: point-based methods, methods and applications, point-based graphics, modelling, computational geometry

ACM CCS: Computational Geometry and Object Modeling → Geometric algorithm

1. Introduction

The Laplace–Beltrami operator (LBO) ($\Delta_{\mathcal{M}}$) plays a critical role in the spectral geometry analysis. It is a fundamental tool and has been intensively applied in numerous geometry processing applications, such as surface filtering [DMSB99], geometry modeling [YZX*04], mesh segmentation [LZ07] and skeleton extraction [TAOZ12]. Developing a robust $\Delta_{\mathcal{M}}$ estimation is thus one of essential geometry processing tasks, and a vast related literature [Tau95, DMSB99, WMKG07, BSW08, CLB*09] has emerged in the past two decades. Nonetheless, most of these works mainly focus on triangular surface models.

As commodity depth sensors become widely available for scanning acquisition, point-based representation is now ubiquitous in computer graphics. There has recently been increasing interest in discretizing $\Delta_{\mathcal{M}}$ on point set surfaces. Belkin *et al.* [BSW09] first proposed a provable $\Delta_{\mathcal{M}}$ (BSW) on point-sampled surfaces, which can converge yet be not symmetrizable. With a local Voronoi cell, a converged and symmetrizable $\Delta_{\mathcal{M}}$ (PBMH) was constructed in [LPG12]. However, their convergence usually degrades in cases with sharp edges or spikes. The $\Delta_{\mathcal{M}}$ proposed by Petronetto *et al.* in [PPH*13] can handle sharp features. It is, however, sensitive to noise and outliers thus not suitable for raw scans.

In this paper, we develop a new $\Delta_{\mathcal{M}}$ defined on point set surfaces, which can handle the shapes containing sharp features. It is

[*]Corresponding author: Yunhai Wang (cloudseawang@gmail.com)

achieved by improving the discretization scheme in [LPG12] with an anisotropic Voronoi diagram to estimate area elements of each point. This new scheme not only allows us to construct the local Voronoi cell for each point on sharp edges, but also provides a more accurate geodesic distance approximation, which is capable of alleviating the $\Delta_{\mathcal{M}}$ sensitivity to the noise. Specifically, to compute the area of the anisotropic Voronoi cell, we conduct an efficient isotropic simplification and provide the convergence proof. Compared to previous approaches [LPG12, PPH*13], our $\Delta_{\mathcal{M}}$ is more accurate and robust, and can deal with noisy raw scans as shown in Figure 13.

We show the effectiveness of the proposed discretization method through a set of comparisons to state-of-the-art techniques. We also demonstrate the performance of our $\Delta_{\mathcal{M}}$ with various applications, such as point-based surface filtering [PG01], reconstruction [LPG12] and skeleton extraction [TAOZ12].

In summary, our main contributions include:

- the development of a generalized anisotropic Voronoi diagram to estimate the volume element for the approximation of the $\Delta_{\mathcal{M}}$ on point set surfaces;
- the proposal of an effective algorithm to compute the area of the anisotropic Voronoi diagram in the local tangential space.

2. Related Work

The Laplace operator Δ is a linear second-order differential operator in Euclidean space $\mathbf{R}^n$, which is defined as the divergence of the gradient. $\Delta_{\mathcal{M}}$ is the more general version of the Laplace operator in the n-dimensional orientable Riemannian manifold $\mathcal{M}$, that is

$$\Delta_{\mathcal{M}} f = \text{div grad} f. \tag{1}$$

In geometry processing, we only consider a smooth two-dimensional (2D) manifold $(\mathcal{M}, g)$ which is embedded in R^3. By applying $\Delta_{\mathcal{M}}$ to a scalar function f, we can get:

$$\Delta_{\mathcal{M}} f = \sum_{i,j=1}^{2} \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi_i} \left(\sqrt{g} g^{ij} \frac{\partial f}{\partial \xi_j} \right), \tag{2}$$

where $g = \det(G)$, $G = [g_{ij}]$ is a metric tensor, and the coefficients g^{ij} are the components of the inverse of G.

According to Equation (2), evaluating $\Delta_{\mathcal{M}}$ requires the estimation of the second derivative, which is not well defined for meshes and point clouds. By using Stokes's theorem to turn a second derivative into two first-order derivatives, $\Delta_{\mathcal{M}}$ can be represented as

$$\int_{\mathcal{M}} \Delta_{\mathcal{M}} f \cdot \phi du = \int_{\mathcal{M}} < \nabla f, \nabla \phi > du, \tag{3}$$

where du is the volume element. However, evaluating first-order derivatives is still a challenge for point clouds. To address this issue, Belkin *et al.* [BN05] propose to approximate $\Delta_{\mathcal{M}}$ with the integration of continuous functions:

$$\Delta_{\mathcal{M}} f(p) = \lim_{t \to 0} \frac{1}{4\pi t^2} \left(\int_{\mathcal{M}} e^{-\frac{||p-q||^2}{4t}} (f(p) - f(q)) \right) du_q \tag{4}$$

which does not require the first and second-order derivatives.

2.1. The discretization schemes

To build the discrete $\Delta_{\mathcal{M}}$, it is necessary to approximate the continuous representation of the $\Delta_{\mathcal{M}}$ for every vertex in meshes or every point in point clouds. Since the topology structure is different between meshes and point clouds, different schemes have been proposed.

Mesh based: The graph Laplacian [Tau95] is the simplest discrete $\Delta_{\mathcal{M}}$ for meshes. Since it is a general extension of the Laplace operator based on finite difference, it can be taken as the approximation of Equation (2). However, it will generate a variety of artifacts such as geometric distortion since it does not consider the geometry information.

To incorporate geometry information, most of the discretization methods rely on the finite element method with different assumptions. Depending on the area element used, some methods may result in non-symmetric matrices with complex eigenvalues and eigenvectors, which limits the application of such $\Delta_{\mathcal{M}}$. Desbrun *et al.* [DMSB99] use the area of the triangles of one-ring neighbours to estimate $\Delta_{\mathcal{M}}$, while the local Voronoi cell is used as the area element for the integration [MDSB03, VL08, DGCD*13]. All of these discrete schemes have similar forms of a *cotangent scheme*, which is widely applied in geometry processing and shape understanding for meshes [BKP*10]. However, it has been shown in [WMKG07] that discrete $\Delta_{\mathcal{M}}$ defined on mesh cannot satisfy all of the properties of its continuous counterpart. In order to guarantee the convergence of the discrete $\Delta_{\mathcal{M}}$, Belkin *et al.* [BSW08] and Chuang *et al.* [CLB*09] construct the discrete schemes based on different continuous functions defined on a 2D manifold, where the former uses the Gaussian kernel, while the latter employs B-Spline as the basis function to define the continuous function in R^3. In both approaches, the area element for the integration calculus is still the original mesh.

To directly implement spectral analysis on meshes, it is necessary to symmetrize a non-symmetric $\Delta_{\mathcal{M}}$ matrix. An empirical symmetrization, which uses $0.5(L + L^T)$ to replace the original non-symmetric $\Delta_{\mathcal{M}}$ matrix L, is proposed in [Lév06] at the expense of partially losing mesh independence. To preserve mesh independence and symmetry of $\Delta_{\mathcal{M}}$ on mesh simultaneously, Vallet and lévy generalize the cotan weights on a discrete exterior calculus framework [VL08].

Point based: Based on different continuous representations, there are different methods to approximate the $\Delta_{\mathcal{M}}$ on point clouds. Deriving from Equation (2), Liang *et al.* [LLWZ12] approximate the $\Delta_{\mathcal{M}}$ using a moving least square method to reconstruct the local surface, while Macdonald *et al.* [MBR11] first compute an embedded function in R^3 based on the closest point method and then apply the standard Cartesian finite difference to get $\Delta_{\mathcal{M}}$. However, both of these two methods require the reconstruction of a local surface for each point, which takes considerable time.

Since the weighted integration schemes (Equation (4)) only depends on the kernel function and the area element, many recent methods are derived from this representation. By using the Gaussian function as the kernel function, Belkin *et al.* [BSW09] perform the integration with triangles generated by local Delaunay triangulation in the local tangent space. Although the resulting $\Delta_{\mathcal{M}}$ is converging as points get denser, the resulting matrix is not guaranteed to be symmetrizable. By improving this approach with the local Vornoni cell to estimate the area element, the $\Delta_{\mathcal{M}}$ matrix generated by Liu *et al.* [LPG12] is guaranteed to be symmetrizable. Since this method requires point clouds to satisfy certain sampling conditions, its convergence will degrade at sharp edges or spikes. With the combination of smoothed particle hydrodynamics and a global optimization procedure to estimate area elements, Petronetto *et al.* [PPH*13] propose a mesh-free discrete $\Delta_{\mathcal{M}}$. Although this $\Delta_{\mathcal{M}}$ is symmetrizable and can handle the shape with sharp features, it is sensitive to noise.

In the paper, our approach focuses on the $\Delta_{\mathcal{M}}$, which can handle the model with sharp features. The key is that we propose a new area element based on an anisotropic Voronoi diagram in the tangential space. Numerical experiments show that this $\Delta_{\mathcal{M}}$ can represent more details in spectral geometry processing.

3. $\Delta_{\mathcal{M}}$ Approximation With Anisotropic Voronoi Diagram

As [BSW09, LPG12], we use the Gaussian function as the kernel function and thus the $\Delta_{\mathcal{M}}$ can be approximated as follows:

$$\Delta_{\mathcal{M}} f(p) = \frac{1}{4\pi t^2} \sum_{q \in p(\delta)} \left(e^{-\frac{||q-p||^2}{4t}} (f(q) - f(p)) vol(q) \right), \quad (5)$$

where $p(\delta)$ consists of neighbour from p within δ, and $vol(q)$ refers to the volume element (area elements in our case) represented by the point q.

In Section 3.1, we introduce the anisotropic Voronoi diagram in the local tangent space to define the area element for every sampled point and then we discretize the anisotropic Voronoi diagram to compute the area element in Section 3.2. Convergence of the $\Delta_{\mathcal{M}}$ is shown in Section 3.3. In Section 3.4, we analyse the reason why our $\Delta_{\mathcal{M}}$ is less sensitive to noise than PBMH [LPG12].

3.1. Anisotropic Voronoi diagram in local tangent space

In Equation (5), the volume element is the only unknown term for $\Delta_{\mathcal{M}}$ approximation. In general, Voronoi cells are used to estimate volume elements for sampled points in Riemannian manifold space or Euclidean space, and are defined as:

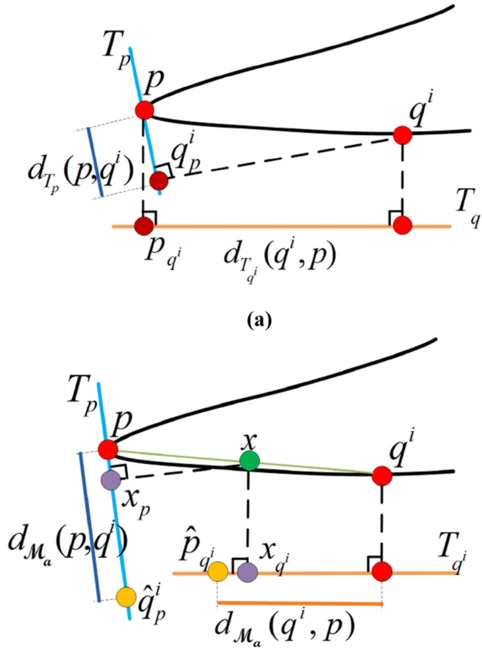
$$Vor_{\mathcal{M}}(p) = \{x \in \mathcal{M} | d_{\mathcal{M}}(p, x) \le d_{\mathcal{M}}(q, x), q \ne p\}, \quad (6)$$

where $d_{\mathcal{M}}(\cdot,\cdot)$ is the geodesic distance between two sampled points.

With the proof that the local Voronoi cell area on the tangential space is converging to its counterpart on the manifold, Liu *et al.* [LPG12] estimate the area of the Euclidean Voronoi cell on the local tangent plane. As a result, the distance $d_{\mathcal{M}}(p, q)$ is approximated by the distance $d_T(p, q)$ in the local tangent space T. Given a point cloud P on the surface $\mathcal{M}$, where the point cloud P is $(\varepsilon, s\varepsilon)$-sampled. It means that the point cloud P satisfies two conditions, that is,

$$\forall x \in, \exists p \in P : ||x - p|| \le \varepsilon,$$

$$\forall p, q \in P : ||p - q|| \ge s\varepsilon, 0 < s < 1.$$



(a)

(b)

Figure 1: *The comparisons between the projections of points on the tangent plane in PBMH [LPG12] and AVD. In PBMH (a), approximative distances $d_{T_p}(p, q^i)$ and $d_{T_{q^i}}(q^i, p)$ of the geodesic distance between the point q and q^i are different in tangent spaces T_p and T_{q^i}. When the local surface at the point p becomes sharper, the ratio $d_{T_p}(p, q^i)/d_{T_q^i}(q^i, p)$ is close to zero, which leads to the degeneration of $\Delta_{\mathcal{M}}$ in PBMH. In contrast, the ratio $d_{\mathcal{M}_a}(p, q^i)/d_{\mathcal{M}_a}(q^i, p)$ of two approximated distances of the geodesic distance $d_{\mathcal{M}}(p, q^i)$ in (b) is equal to* 1*, and x is the middle point of the line pq^i.*

For any point $p \in P$, its Voronoi cell is defined as:

$$Vor(p) = \{x_p \in T_p | d_{T_p}(p, x) \le d_{T_p}(x, q), ||p - q|| \le \delta\}, \quad (7)$$

where x_p is the projection of x on the tangent plane T_p, and $d_{T_p}(\cdot,\cdot)$ is the distance between the projections of two points in the tangent plane T_p. To ensure that the estimated tangent plane is converging to the real tangent plane [BSW09], δ is set to be 10ε. In other words, x_p is selected by projecting all sampled points x to the local tangent plane T_p.

However, this definition has the result that the convergence of the $\Delta_{\mathcal{M}}$ degrades at sharp features. The point p in Figure 1(a) is an example, where $\{q^i | q^i \in P, ||p - q^i|| \le \delta\}$ are the neighbours of p. The distances between p and q^i in the tangent planes T_p and T_{q^i} are

$$d_{T_p}(p, q^i) = \sqrt{||pq^i||^2 - (pq^i \cdot N_p)^2},$$

$$d_{T_{q^i}}(q^i, p) = \sqrt{||pq^i||^2 - (pq^i \cdot N_{q^i})^2}, \quad (8)$$

where T_{q^i} is the tangent plane of q^i, N_p and N_q^i are the normals of p and q^i, respectively. As the local surface at the point p becomes sharper, $|N_p \cdot N_{q^i}|$ will close to zero, while $d_{T_p}(p, q^i)$ will become smaller and $d_{T_{q^i}}(q^i, p)$ will become larger. In other words, $d_{T_p}(p, q^i)/d_{T_{q^i}}(q^i, p)$ will be close to 0. Nonetheless, $d_{T_p}(p, q^i)$ and $d_{T_{q^i}}(q^i, p)$ both approximate the geodesic distance between the point p and q^i, a symmetric distance indicates that $d_{T_p}(p, q^i)/d_{T_{q^i}}(q^i, p)$ should be close to 1. That is the reason why the convergence of the $\Delta_{\mathcal{M}}$ proposed in [LPG12] degrades at sharp edges.

To overcome this drawback, we propose to estimate the volume element of point p with the anisotropic Voronoi diagram [LS03] defined on the local tangent space T_p:

$$Vor_{ani}(p) = \left\{ x_p \in T_p | d_{T_p}(p, x) \le d_{T_{q^i}}(q^i, x) \right\}, \tag{9}$$

where $d_{T_{q^i}}(q^i, x)$ is the distance between the projections of two points q^i and x in the tangent plane T_{q^i}, and x_p is the projection of the point x in the tangent plane T_p. Compared with Equation (7), we can see that all sampled points x are first projected to two tangent planes T_p and T_{q^i} and then x_p is adaptively determined by first finding a proper x through comparing $d_{T_p}(p, x)$ with $d_{T_{q^i}}(q^i, x)$. This implies that the approximated distance $d_{\mathcal{M}_a}(p, q^i)$ of the geodesic distance between two neighbouring points p and q^i is not $d_{T_p}(p, q^i)$, and also not $d_{T_q^i}(q^i, p)$, but

$$\begin{aligned} d_{\mathcal{M}_a}(p, q^i) &= d_{T_p}(p, \hat{x}) + d_{T_{q^i}}(\hat{x}, q^i) \\ &\approx d_{\mathcal{M}}(p, \hat{x}) + d_{\mathcal{M}}(\hat{x}, q^i) \\ &= d_{\mathcal{M}}(p, q^i), \end{aligned} \tag{10}$$

where the point $\hat{x}$ is the point located at the geodesic line between p and q^i. Furthermore, in terms of Lemma 1.5 [LPG12], the approximate distance holds $1 \le \frac{d_{\mathcal{M}}(p,q^i)}{d_{T_p}(p,\hat{x})+d_{T_{q^i}}(\hat{x},q^i)} \le 1 + O(\varepsilon^2/\rho^2)$, where ρ is the smaller of the distances from the point p and q^i to the medial axis. Meanwhile, we select $\hat{x}$ to hold $d_{T_p}(p, \hat{x}) = d_{T_{q^i}}(\hat{x}, q^i)$, so that $d_{\mathcal{M}_a}(p, q^i)/d_{\mathcal{M}_a}(q^i, p)$ is equal to 1. It can avoid the issue that $d_{T_p}(p, q^i)/d_{T_{q^i}}(q^i, p)$ is close to 0 or ∞ at the sharp feature point [LPG12]. Thus, it prevents the degeneration of the $\Delta_{\mathcal{M}}$ at the sharp feature point. For convenience, we refer to our approach as AVD.

3.2. Computing Voronoi cell

Although the anisotropic distance measure implied by the anisotropic Voronoi diagram is more accurate, it is difficult to estimate the area of such a Voronoi cell in the local tangent space. The boundary of the Voronoi cell $Vor_{ani}(p)$ is composed of patches of a quadratic curve as shown in [LS03]. The complexity of computing an anisotropic Voronoi diagram is in $O(n^2)$. Furthermore, the complexity of computing the area of the Voronoi cell composed of quadratic curves is higher than that of computing the area of the Voronoi cell composed of lines. To facilitate the local Voronoi cell computation, we propose to convert the anisotropic Voronoi cell to an isotropic version while the distance is preserved.

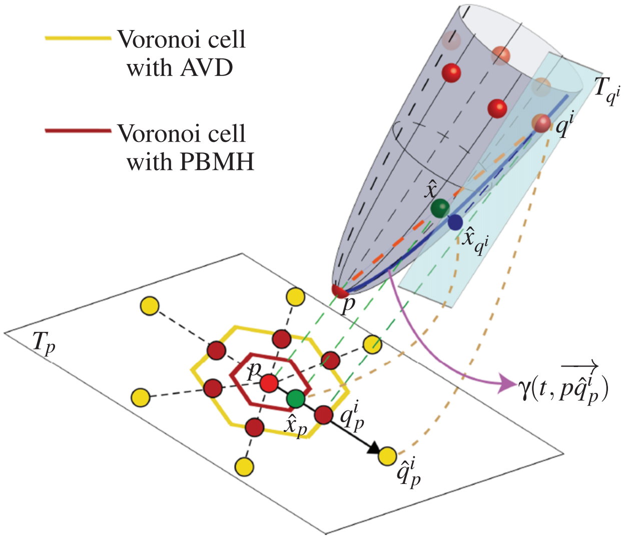


Figure 2: *Illustration of Voronoi cell computation. $\{q^i\}$ are neighbours of the point p. T_p and T_{q^i} are the tangent space of the points p and q^i separately. $\hat{x}_p$ and q^i_p are the orthogonal projection of the points $\hat{x}$ and q^i on the local tangent space T_p. $\hat{x}_{q^i}$ is the orthogonal projection of the point $\hat{x}$ on the local tangent space T_{q^i}. $\hat{q}^i_p$ is the approximate inverse exponential map of the point q^i on the local tangent space T_p.*

Suppose neighbours $\{q^i\}$ of the point p within δ away from p are considered for the Voronoi cell estimation in Figure 2. We represent the parameterized geodesic as $\gamma(t, \overrightarrow{p\hat{q}^i_p}), t \in (-\epsilon, \epsilon)$ with initial conditions $\gamma(0, \overrightarrow{p\hat{q}^i_p}) = p$ and $\gamma(0, \overrightarrow{p\hat{q}^i_p}) = \overrightarrow{p\hat{q}^i_p}$. We firstly project $\{q^i\}$ on the local tangent space of the point p via an approximate inverse exponential map,

$$exp_p\left(\overrightarrow{p\hat{q}^i_p}\right) = \gamma\left(1, \overrightarrow{p\hat{q}^i_p}\right) \approx q^i, \tag{11}$$

where the length of the vector $\overrightarrow{p\hat{q}^i_p}$ is equal to the approximated geodesic distance between p and q^i, that is,

$$||\overrightarrow{p\hat{q}^i_p}|| = d_{T_p}(p, \hat{x}) + d_{T_{q^i}}(\hat{x}, q^i), \tag{12}$$

the direction of the vector $\overrightarrow{p\hat{q}^i_p}$ is same with the direction of the vector $\overrightarrow{pq^i_p}$, and q^i_p is the orthogonal projection of the point q^i on the local tangent space T_p. Then, we take the point $\hat{q}^i_p$ to construct the isotropic Voronoi cell of the point p like PBMH [LPG12], as shown in Figure 2. To reduce the time cost of searching for $\hat{x}$, the point $\hat{x}$ is selected as the middle point between the points p and q^i in our approach. Compared with PBMH [LPG12], the proposed Voronoi cell preserves the symmetry of the distance in different tangent spaces. Furthermore, in Theorem 3.1, we prove that when the point $\hat{x}$ is the midpoint between p and q^i, then the area of the estimated Voronoi cell converges to the area of the Voronoi cell in the manifold.

3.3. Convergence theorem

The proof of the convergence of $\Delta_{\mathcal{M}}$ in PBMH [LPG12] consists of two steps: proving the convergence of the estimated Voronoi cell area and proving the convergence of $\Delta_{\mathcal{M}}$ based on the Voronoi cell area convergence result. Since the difference between our AVD and PBMH [LPG12] is that different area elements are used, it is only necessary to prove that the estimated area of the Voronoi cell is converged to the real Voronoi cell area in our $\Delta_{\mathcal{M}}$ (Theorem 3.1). The proofs for the convergence of the Voronoi cell area are provided in the Appendix.

Theorem 1 (Voronoi cell approximation)**.** *Consider the Voronoi cell of the point $p \in P$ where P is a $(\varepsilon, s\varepsilon)$-sample of the manifold $\mathcal{M}$, the Voronoi cell $Vor_{ani}(p)$ on the tangent space T_p built by our algorithm, and the Voronoi cell $Vor(p)$ on the tangent space T_p built by PBMH [LPG12],*

$$\left|\left|\frac{vol(Vor_{ani}(p))}{vol(Vor(p))} - 1\right|\right| \leq O(\varepsilon/\rho) \tag{13}$$

$$\left|\left|\frac{vol(Vor_{\mathcal{M}}(p))}{vol(Vor_{ani}(p))} - 1\right|\right| \leq O(\varepsilon/\rho) \tag{14}$$

holds when ε is small enough.

3.4. Noise analysis

The scanned point clouds often contain different degrees of noise. It is assumed that the point q^i is moved to $q^{i'}$ with ω along the normal $\mathbf{n}_{q^i}$ as shown in Figure 3, $q^{i'} = q^i + \omega\mathbf{n}_{q^i}$. The error of the projection distance pq^i in the tangent space T_p generated by PBMH [LPG12] (Figure 3a) is:

$$\left|\left|pq_p^i - pq_p^{i'}\right|\right| = \left|\left|q_p^i q_p^{i'}\right|\right| = \omega\sqrt{1-(\mathbf{n}_p\cdot\mathbf{n}_{q^i})^2}. \tag{15}$$

In terms of Equation (12), the projection distances of pq^i and $pq^{i'}$ in the tangent space T_p are $||px_p|| + ||x_{q^i}\hat{q}_p^i||$ and $||px'_p|| + ||x_{q^i}\hat{q}_p^{i'}||$, respectively (Figure 3b). Thus, the error of the projection distance of pq^i in AVD is

$$\left|\left|q_p^{i'} q_p^i\right|\right| = \frac{\omega}{2}\sqrt{1-(\mathbf{n}_p\cdot\mathbf{n}_{q^i})^2}. \tag{16}$$

From Equation (15) and Equation (16), we can see the error generated by our approach is half of the one of PBMH. Thus, our $\Delta_{\mathcal{M}}$ is less sensitive to noise compared to PBMH.

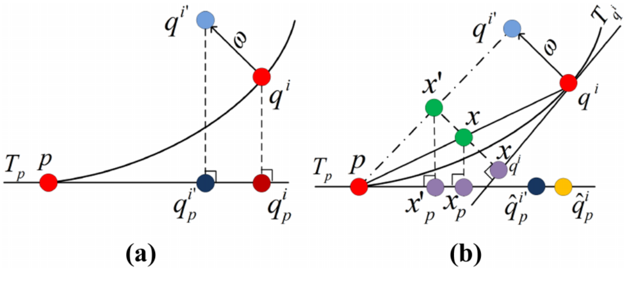


Figure 3: *The influence of noise on the distance measure used in PBMH [LPG12](a) and our AVD(b), where ω is the normal shift of the point q^i, x and x' are the middle points of pq^i and $pq^{i'}$. The error $||\hat{q}_p^{i'}\hat{q}_p^i||$ generated in (b) is half of the error $||q_p^{i'}q_p^i||$ in (a).*

3.5. Normal estimation

Normal plays a very important role in the proposed algorithm to define the tangent space. When normal vectors are not available, it is necessary to estimate normal vectors. The estimation of normals can be reduced to the eigenvector problem of the covariance matrix in principal component analysis (PCA) [PGK02]. However, PCA is sensitive to noise and outliers for normal estimation. To get more robust and accurate normal vectors, we applied a robust, iterative and convergent mean shift filter to refine normal vectors estimated in PCA. More details can be found in the paper [HJY08].

4. Results and Applications

In this section, we verified the effectiveness of our $\Delta_{\mathcal{M}}$ with various point clouds data sets. All experiments were conducted on a Windows platform with Intel Core 2 Duo 2.6GHz CPU and 2GB RAM. In our implementation, the generation of the $\Delta_{\mathcal{M}}$ matrix were written in C++, and the eigenproblem was solved in MATLAB. Since real scanned point clouds may be not $(\varepsilon, s\varepsilon)$-sampled, the neighbours of the point p was selected as the union of δ-neighbours and k-neighbours. In the experiments, k was selected as 20.

4.1. Accuracy evaluation

The estimation for the area element at every sampled point is the key for the approximation of the $\Delta_{\mathcal{M}}$ on point clouds. Our first experiment was to verify whether the area estimated by our approach is more accurate than others with two uniformly sampled models: one sphere and one cone. Since we have the ground truth area $Area_c$ of these models, we measured the error of the estimated area by

$$Area\ Errors = \frac{|Area_c - \sum_{p\in P} vol(p)|}{Area_c}, \tag{17}$$

where p is a point in sampled point clouds P and $vol(p)$ refers to the estimated area of the constructed Vornoni cell around p. Figure 4 shows the comparison between PBMH [LPG12] and our approach. We can see that the estimated surface area generated by our approach is closer to the area of the continuous surface than the area generated by PBMH [LPG12].

In order to evaluate the accuracy of our $\Delta_{\mathcal{M}}$, we first defined several analytic functions over the unit sphere and then compared computed Laplacians with the analytic solutions. The sphere was sampled with three different settings: uniform sampling, non-uniform sampling and noise perturbation, where 5% of Gaussian noise was added in the normal and tangent components of the point clouds. Similar to Petronetto *et al.* [PPH*13], we measured the accuracy with L_2 error and L_∞ error for the functions $f(x,y,z) = x, x^2, e^x$,

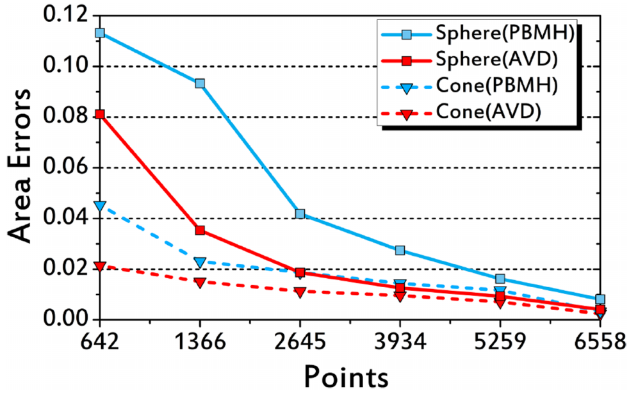


Figure 4: *The comparisons of surface area estimation on the uniform sampled sphere and cone with PBMH [LPG12] and our approach.*

Table 1: *Approximation error of $\Delta_{\mathcal{M}} f$ on the uniformly sampled sphere (8k points) using L_2 error and L_∞ error.*

		Uniform	
		L_2 Error	L_∞ Error
$f(x,y,z) = x$	PBMH	12.64	1.58
	AVD	11.6	1.36
$f(x,y,z) = x^2$	PBMH	5.69	0.97
	AVD	4.93	0.81
$f(x,y,z) = e^x$	PBMH	11.02	0.75
	AVD	9.15	0.66

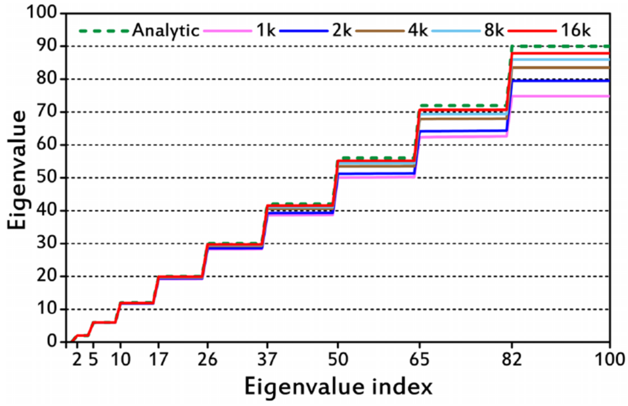


Figure 5: *Convergence of the eigenvalues of our $\Delta_{\mathcal{M}}$ on the unit sphere with different number of sampling points.*

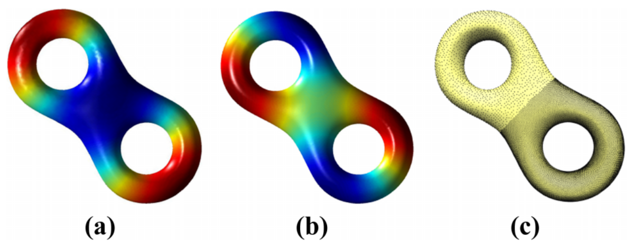


Figure 6: *The eigenvectors of H^2 (a), and H^4 (b) of the $\Delta_{\mathcal{M}}$ proposed by our approach, for the symmetric model 'Eight' (40k points) (c) with the non-uniform sampling distribution.*

which are shown in Table 1. We can see that both L_2 error and L_∞ error with our $\Delta_{\mathcal{M}}$ are less than PBMH.

4.2. Convergence of $\Delta_{\mathcal{M}}$

To evaluate the convergence of $\Delta_{\mathcal{M}}$ in our approach, we evaluated eigenvalues of $\Delta_{\mathcal{M}}$ on the unit sphere with a different number of uniform sampling points, as shown in Figure 5. We can see that the eigenvalues of $\Delta_{\mathcal{M}}$ generated by our approach tend to the analytical eigenvalues as the number of sample points increase. Moreover, the multiplicity of the eigenvalues with different sample points matches closely the multiplicity of their analytical counterpart.

4.3. Non-uniformly sampled and noisy point clouds

To verify the robustness of the proposed operator for non-uniformly sampled point clouds, we first conducted an experiment on the 'symmetric' two-hold torus ('Eight') model, where two handles were sampled with different sampling rates. As shown in Figure 6, the eigenvectors H^2 and H^4 of the $\Delta_{\mathcal{M}}$ are still symmetric over the surface, although the point distribution is non-uniform.

We then conducted another experiment to demonstrate the effectiveness of our $\Delta_{\mathcal{M}}$ in handling noisy point clouds. Figure 7 shows the eigenvalues of the $\Delta_{\mathcal{M}}$ on the unit sphere under three settings: uniform sampling, non-uniform sampling and noise perturbations, where 5% of Gaussian noise was added in the normal and tangent components of the point clouds. It showed that the eigenvalues generated by our approach are closer to the analytical eigenvalues than PBMH [LPG12]. The stability of the eigenvectors of the $\Delta_{\mathcal{M}}$ is illustrated in the third row of Figure 7, where we can see that the eigenvectors H^4 are almost the same under three settings.

4.4. Sharp features

To illustrate the efficiency of the proposed operator in handling the models with sharp features, Figure 8 compares the eigenvectors of $\Delta_{\mathcal{M}}$ on the model with sharp features under isometric transformations. Here, we used a flat square model sampled with 101 × 101 points. The isometric models with sharp features were created by folding the flat square with 0, 30, 70 and 120 degrees from left to right in Figure 8. Notice that the eigenvectors of the $\Delta_{\mathcal{M}}$ on BSW [BSW09] and PBMH [LPG12] vary under isometric transformation, while the eigenvectors of the proposed operator are well preserved. Furthermore, Figure 9 shows asymptotic behaviour in the semi-log scale for our approach on the model with sharp features when discretizing the $\Delta_{\mathcal{M}}$ in a uniformly sampled folding square with increased density. Notice that the error resulting from the AVD discretization was always less than the one obtained with BSW and PBMH. Thus, the proposed operator is robust and stable for the model with sharp features. In the next subsection, we also

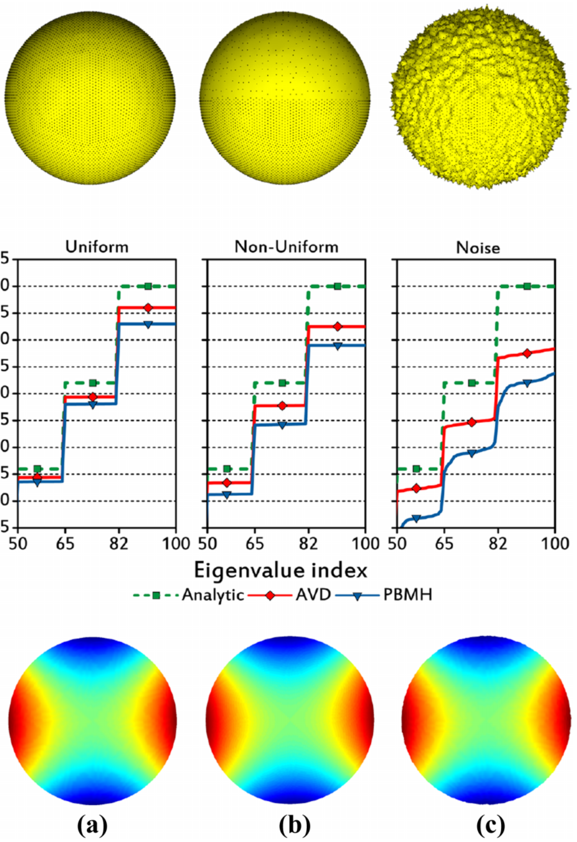


Figure 7: *The effectiveness of our $\Delta_{\mathcal{M}}$ in handling a uniformly sampled sphere, non-uniformly sampled sphere and noisy sphere. The first row shows the sampled point clouds on these three spheres from left to right, the second row shows the eigenvalues of $\Delta_{\mathcal{M}}$ generated by our AVD, PBMH and the analytic solutions, and the last row shows the eigenvectors H^4 of the $\Delta_{\mathcal{M}}$ under three settings.*

proved the capacity of our approach to better encode sharp features in surface reconstruction 11 and skeleton extraction 12.

4.5. Application

To verify whether the $\Delta_{\mathcal{M}}$ estimated with our approach can represent more details with sharp features, we used the $\Delta_{\mathcal{M}}$ for the spectral representation of point clouds. First, the eigen problem of the $\Delta_{\mathcal{M}}$ matrix was solved to get the base vectors H^i of the sampled surface [LPG12]. Then, the base vectors were used to reconstruct point clouds:

$$f = \sum < f, H^i > H^i. \tag{18}$$

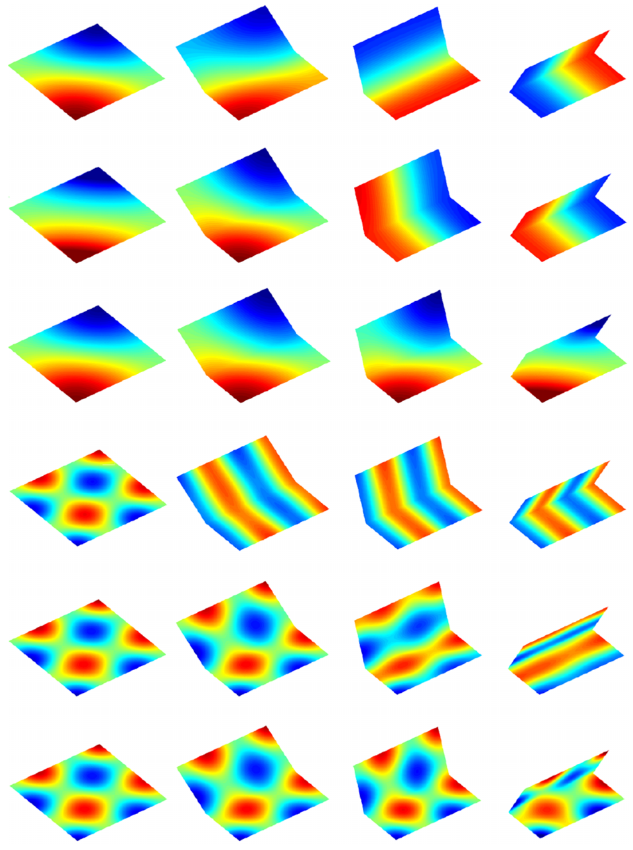


Figure 8: *The comparisons of eigenvectors H^2 (the first three lines) and H^{10} (the last three lines) on the model with sharp features. The first and fourth lines are the results generated by BSW [BSW09], the second and fifth lines are the results generated by PBMH [LPG12]. The third and sixth lines are the results generated by our approach.*

Figure 11 and Figure 10 qualitatively and quantitatively compared the reconstruction of point clouds generated by other approaches and our $\Delta_{\mathcal{M}}$, respectively. In Figure 11, we used separate $2k$ bases with the largest eigenvalues to reconstruct the hand model in our approach, BSW [BSW09] and PBMH [LPG12]. Compared with PBMH and our AVD, noise is introduced into the model reconstructed in BSW [BSW09]. More geometric details with sharp features are preserved in our approach than in PBMH [LPG12] and less reconstruction error is generated in our approach than in PBMH. The larger the eigenvalue is, the lower the frequency information that is encoded in the corresponding base. The experiment, therefore, showed that more sharp features are encoded in our approach than in PBMH [LPG12]. Furthermore, we used all bases to reconstruct models to demonstrate the accuracy of surface representation in our approach. Figure 10 shows the max error and the mean error in our approach and PBMH [LPG12] between the original models and the reconstructed models with all bases, where we can see that the max error was becoming less and less as the number of used bases was increasing. Meanwhile, both the max error and the mean error generated by our $\Delta_{\mathcal{M}}$ are less than by PBMH [LPG12]. We can conclude that a more highly accurate surface is generated by our approach than by PBMH [LPG12].

To further verify that our approach is efficient in representing the geometry structure of point cloud data, we configured our $\Delta_{\mathcal{M}}$ and PBMH to the point clouds-based skeleton extraction approach [TAOZ12] and see which extracted skeleton captures more geometry structure. As shown in Figure 12, we applied AVD and PBMH

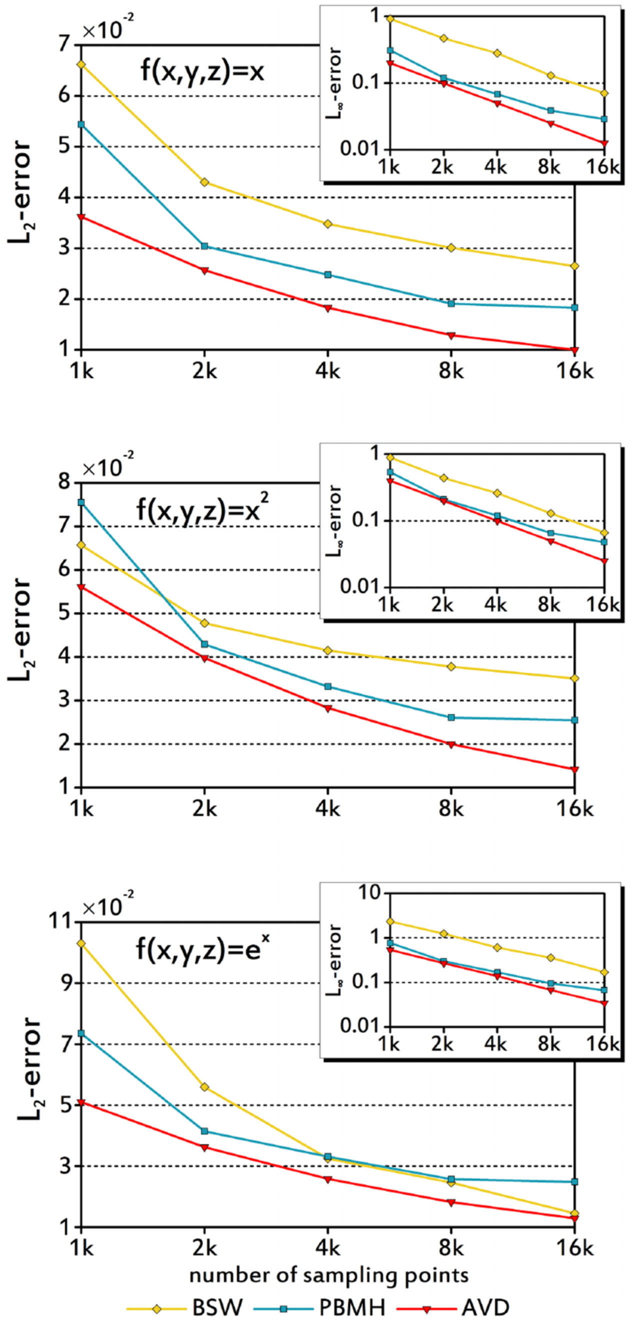
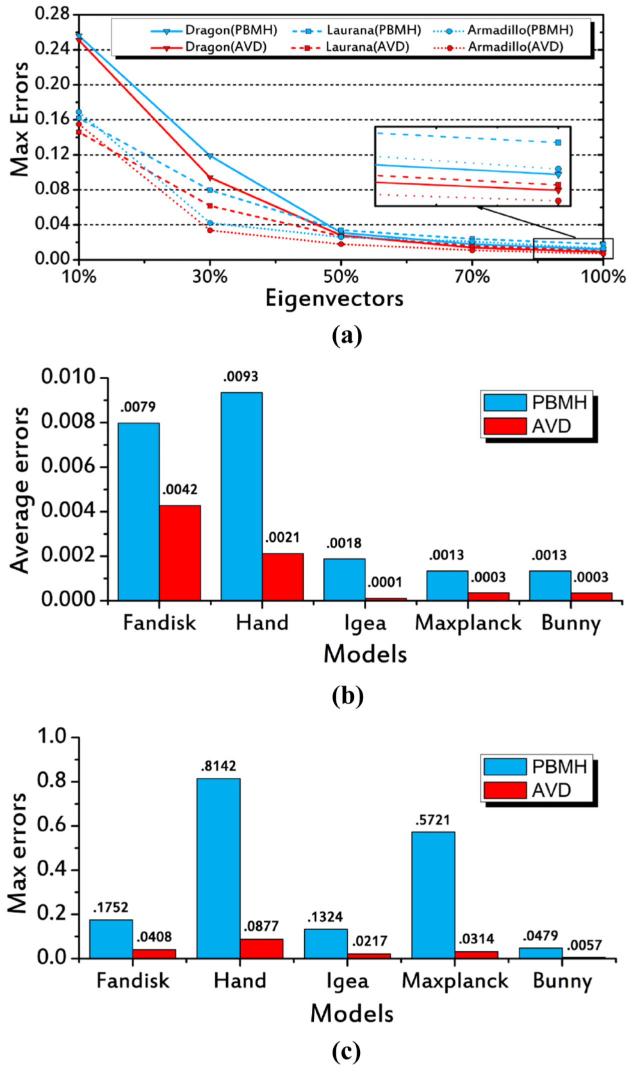


Figure 9: *Consistency analysis of $\Delta_{\mathcal{M}} f$ on uniformly sampled folding square with 70°. The curves allow to compare the AVD-based method against BSW [BSW09] and PBMH [LPG12], using average error in l^2 (larger graphs) and l^∞ (top right) norms.*



(a)

(b)

(c)

Figure 10: *The comparison of errors of reconstructed surfaces with PBMH and our AVD. (a) The max errors of the reconstructed surface with different numbers of base vectors, the bigger the eigenvalue corresponding to the base is, the earlier the base is used for surface reconstruction; (b) the mean errors of reconstructed surface with all base vectors; (c) the max errors of reconstructed surface with all base vectors.*

separately to extract skeletons, which are represented with the same number of points. Figures 12(a) and (c) show that skeletons extracted with PBMH mainly captured the straight line branch and miss a few structures, such as the ankles of the horse model (Figure 12a). Figures 12(b) and (d) show that skeletons extracted with AVD can capture more geometry structures with sharp features, such as the curve branch of the skeleton in Figure 12, than with PBMH. This further proved that our approach is more effective in high level understanding of point clouds data than PBMH [LPG12], and it especially better encodes sharp features in our $\Delta_{\mathcal{M}}$.

Finally, we applied our approach to estimate the $\Delta_{\mathcal{M}}$ of the scanned point clouds. Multi-scale models were reconstructed with a different number of base vectors of the estimated $\Delta_{\mathcal{M}}$ shown in Figure 13. Although the scanned point clouds are full of holes, noise and irregular points, we found that our approach for the estimation of the $\Delta_{\mathcal{M}}$ is effective.

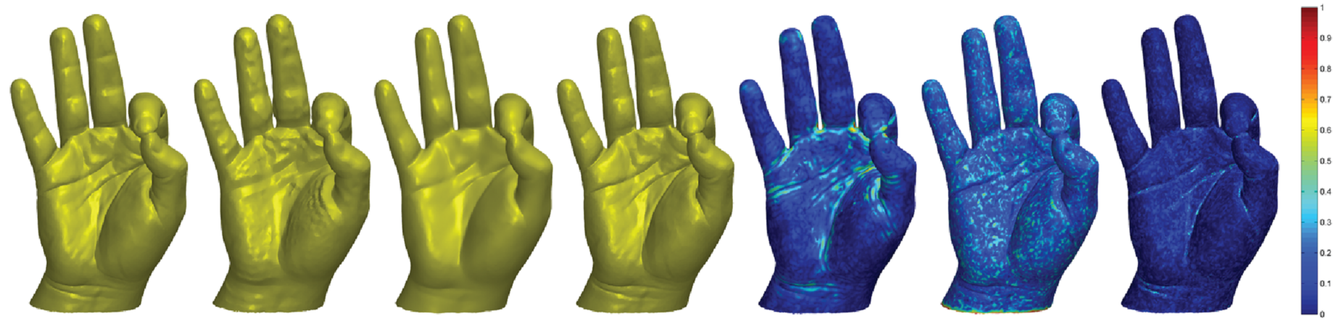


Figure 11: *The reconstruction of point clouds data based on bases. From left to right, we show the original hand model (7k points), the reconstruction results with 2k bases corresponding to the largest eigenvalues on BSW, PBMH and our AVD, and corresponding errors between the reconstructed models and the original model in BSW, PBMH and our AVD.*

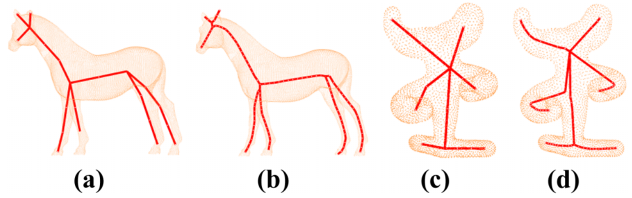
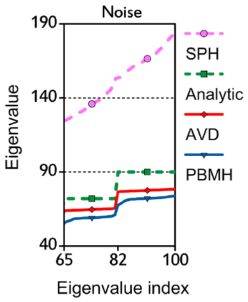


Figure 12: *The comparison of the extracted skeleton. (a) and (c) are the extracted skeletons with PBMH [LPG12], respectively; (b) and (d) are the extracted skeletons with our AVD, respectively.*

4.6. Comparison with mesh-free $\Delta_{\mathcal{M}}$



Recently, Petronetto *et al.* [PPH*13] proposed another $\Delta_{\mathcal{M}}$, SPH, which is a mesh-free version. In addition to the fact that different kernel functions are used to define continuous functions on surfaces, the main difference with our approach lies in the area estimation. They regarded the area estimation as a global optimization problem, which is quite slow for large data. Figure 14 shows the comparison of estimated areas and running times between SPH [PPH*13] and our approach. It can be noticed that the accuracy of the estimated area in our approach is comparable with SPH, while the advantage of our approach in time cost is more obvious than SPH with the increasing of the number of points. Finally, SPH is not as stable under noise perturbations as our $\Delta_{\mathcal{M}}$. By comparing the eigenvalues of the $\Delta_{\mathcal{M}}$ with the analytic solutions over the noisy sphere shown in Figure 7, our $\Delta_{\mathcal{M}}$ is the most robust to noise compared to SPH and PBMH (see inset).

5. Conclusion and Future Work

To approximate the $\Delta_{\mathcal{M}}$ over point clouds, we propose to use an anisotropic Voronoi diagram to estimate the volume element for every sampled point. In order to facilitate the local Voronoi cell computation, we introduce an approximate method that converts an anisotropic Voronoi cell to an isotropic one. Compared to existing approaches, the proposed $\Delta_{\mathcal{M}}$ not only provides more accurate estimation for the volume element but also is less sensitive to noise and prevents the degeneration at sharp edges or spikes. Finally, we demonstrate its effectiveness and usefulness with several spectral analysis and processing applications.

There are also limitations with our current approach. First, our approach is based on Voronoi cell estimation. It is a challenging problem to accurately estimate the Voronoi cell area near the boundary when there is a boundary on the surface. Thus, the convergence of our approach is the same as PBMH [LPG12] degrades at the

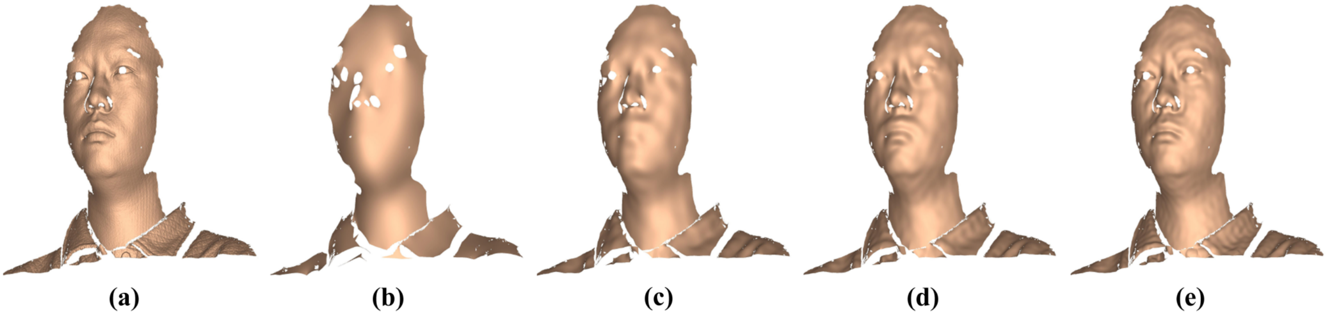


Figure 13: *The representation of real scanned point clouds (311k points) based on different numbers of bases. (a) Original point clouds; (b)–(e) The reconstruction based on different number of bases: 100 (b), 500 (c), 1000 (d) and 1500 (e).*

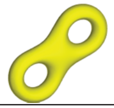
Model				
Points		12k	15k	20k
Area	Triangles	3.98	5.56	5.52
	SPH	4.03	5.50	5.37
	AVD	4.06	5.60	5.59
Error	SPH	1.26%	1.08%	2.72%
	AVD	2.01%	0.72%	1.27%
Times	SPH	31s	151s	505s
	AVD	38s	55s	88s

Figure 14: *Comparisons of estimated areas and running times for calculating the area element between SPH [PPH*13] and our AVD.*

boundary points. The duplicate points are singularities for the Voronoi cell estimation, and resampling is a necessary preprocessing to estimate $\Delta_{\mathcal{M}}$ at the case. Second, accurate normal estimation also plays an important role in our approach. When there are outlier points or too much noise in point clouds, the accuracy of the proposed $\Delta_{\mathcal{M}}$ is affected by the error of estimated normal. In the future, we will investigate more robust algorithms to improve the accuracy of the $\Delta_{\mathcal{M}}$ for these cases. We would also like to extend this $\Delta_{\mathcal{M}}$ to high-dimensional data and investigate more interesting applications.

Acknowledgements

The authors wish to thank the anonymous reviewers for their remarks that helped us improving the paper. This work is partly supported by the grants of Natural Science Foundation of China (61772097), NSFC Guangdong Joint Fund (U1401252, U1501255), the National Key Research & Development Plan of China (2016YFB1001404), Science Challenge Project, No. TZ2016002, Guangdong Science and Technology Program (2015A030312015), Shandong Provincial Natural Science Foundation (2016ZRE27617) and the Fundamental Research Funds of Shandong University.

Appendix A: Convergence Proof

In construction of our LBO, the assumption is that we have a continuous differentiable Riemannian manifold $\mathcal{M}$ on which the sample set P lies. We are going to prove that our estimation of the Voronoi cell area $vol(Vor_{ani}(p))$ is converging to the real Voronoi cell area $vol(Vor_{\mathcal{M}}(p))$ as point clouds get denser.

Lemma. (Voronoi cell approximation). Consider the Voronoi cell of the point $p \in P$ where P is a $(\varepsilon, s\varepsilon)$-sample of the manifold $Vor_{\mathcal{M}}(p)$, the Voronoi cell on the tangent space T_p $Vor_{ani}(p)$ built by our algorithm, and the Voronoi cell on the tangent space T_p $Vor(p)$ built with Liu et al.'s algorithm [LPG12],

$$\left|\left|\frac{vol(Vor_{ani}(p))}{vol(Vor(p))} - 1\right|\right| \le O(\varepsilon/\rho), \tag{A.1}$$

$$\left|\left|\frac{vol(Vor_{\mathcal{M}}(p))}{vol(Vor_{ani}(p))} - 1\right|\right| \le O(\varepsilon/\rho) \tag{A.2}$$

holds when ε is small enough.

Proof. Suppose $p \cup q^i_p, q^i \in p(\delta)$ be influencing points of the Voronoi cells $Vor(p)$, where q^i is the orthogonal projection of the point q^i on the tangent space T_p. q^i_p can be represented as:

$$q^i_p = p + ||\vec{pq}||\cos\alpha_i \frac{\vec{pq^i_p}}{||\vec{pq^i_p}||}. \tag{A.3}$$

In terms of construction of the Voronoi cell $Vor_{ani}(p)$, influencing points of the Voronoi cell $Vor_{ani}(p)$ are $p \cup \hat{q}^i_p$:

$$\hat{q}^i_p = p + \frac{||\vec{pq}||}{2}(\cos\alpha_i + \cos\beta_i)\frac{\vec{pq^i_p}}{||\vec{pq^i_p}||}, \tag{A.4}$$

where α_i is the angle between the vector $\vec{pq^i}$ and the tangent spaces T_p, β_i is the angle between the vector pq^i and the tangent spaces T^i_q, and $0 \le \alpha_i, \beta_i \le \frac{\pi}{2}$. By subtracting Equation (A.3) from Equation (A.4), we can have

$$\begin{aligned} t &= \hat{q}^i_p - q^i_p \\ &= \frac{||\vec{pq}||}{2}(\cos\beta_i - \cos\alpha_i)\frac{\vec{pq^i_p}}{||\vec{pq^i_p}||} \\ &= -||\vec{pq}||\sin\left(\frac{\alpha_i+\beta_i}{2}\right)\sin\left(\frac{\beta_i-\alpha_i}{2}\right)\frac{\vec{pq^i_p}}{||\vec{pq^i_p}||}. \end{aligned} \tag{A.5}$$

When $\beta_i \ge \alpha_i$, we have

$$t \ge -||\vec{pq}||\sin\left(\frac{\beta_i-\alpha_i}{2}\right) \tag{A.6}$$

on Equation (A.5). According to Lemma 2.1 in the supplementary appendices of [LPG12], we have $\beta_i - \alpha_i \le O(\varepsilon/\rho)$. Thus, we have

$$t \ge -||\vec{pq}||O(\varepsilon/\rho). \tag{A.7}$$

When $\beta_i < \alpha_i$, we can obtain

$$t \le ||\vec{pq}||O(\varepsilon/\rho) \tag{A.8}$$

in a similar way.

Let $\hat{q} = p + (q^i_p - p)t'$. By combining Equation (A.3) with Equation (A.7) and Equation (A.8), we have

$$1 - \frac{O(\varepsilon/\rho)}{\cos\alpha_i} \le t' \le 1 + \frac{O(\varepsilon/\rho)}{\cos\alpha_i}. \tag{A.9}$$

According to lemmas 3.1, 3.2 and 3.3 in the supplementary appendix of [LPG12], we have

$$\left(min\left(1-\frac{O(\varepsilon/\rho)}{\cos\alpha_i}\right)\right)^2 \le \frac{vol(Vor_{ani}(p))}{vol(Vor(p)} \le \left(max\left(1+\frac{O(\varepsilon/\rho)}{\cos\alpha_i}\right)\right)^2. \tag{A.10}$$

When point clouds get denser, we have $\alpha_i < \pi/4$, $\beta_i < \pi/4$. Thus,

$$\left\|\frac{vol(Vor_{ani}(p))}{vol(Vor(p))}-1\right\| \le O(\varepsilon/\rho). \tag{A.11}$$

According to theorem 4.1 in [LPG12],

$$\left\|\frac{vol(Vor_{\mathcal{M}}(p))}{vol(Vor(p))}-1\right\| \le O(\varepsilon^2/\rho^2). \tag{A.12}$$

By combining Equation (A.11) and Equation (A.12), we have

$$\left\|\frac{vol(Vor_{ani}(p))}{vol_{\mathcal{M}}(Vor(p))}-1\right\| \le O(\varepsilon/\rho). \tag{A.13}$$

□

References

[BKP*10] Botsch M., Kobbelt L., Pauly M., Alliez P., Lévy B.: *Polygon Mesh Processing*. CRC Press, Boca Raton, FL, USA, 2010.

[BN05] Belkin M., Niyogi P.: Towards a theoretical foundation for Laplacian-based manifold methods. In *18th Annual Conference on Learning Theory*. P. Auer and R. Meir (Eds.). Springer, Berlin, Heidelberg, 2005, pp. 486–500.

[BSW08] Belkin M., Sun J., Wang Y.: Discrete laplace operator on meshed surfaces. In *Proceedings of the Twenty-Fourth Annual Symposium on Computational Geometry* (New York, USA, 2008), ACM, pp. 278–287.

[BSW09] Belkin M., Sun J., Wang Y.: Constructing Laplace operator from point clouds in $\mathbf{R}^d$. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms* (Philadelphia, PA, USA, 2009), Society for Industrial and Applied Mathematics, pp. 1031–1040.

[CLB*09] Chuang M., Luo L., Brown B. J., Rusinkiewicz S., Kazhdan M.: Estimating the Laplace-Beltrami operator by restricting 3d functions. *Computer Graphics Forum 28* (2009), 1475–1484.

[DGCD*13] De Goes F., Crane K., Desbrun M., Schröder P., et al.: Digital geometry processing with discrete exterior calculus. In *ACM SIGGRAPH 2013 Courses* (New York, USA, 2013), ACM, p. 7.

[DMSB99] Desbrun M., Meyer M., Schröder P., Barr A. H.: Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, USA, 1999), ACM Press/Addison-Wesley Publishing Co., pp. 317–324.

[HJY08] Hongxing Q., Jie Y., Yuemin Z.: Nonuniform bilateral filtering for point sets and surface attributes. *The Visual Computer 24* (2008), 10673–1074.

[Lév06] Lévy B.: Laplace-Beltrami eigenfunctions towards an algorithm that 'understands' geometry. In *IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06)* (Piscataway, NJ, 2006), S. Kawada (Ed.), IEEE, pp. 13–20.

[LLWZ12] Liang J., Lai R. Wong T. W. Zhao H.: Geometric understanding of point clouds using Laplace-Beltrami operator. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Piscataway, NJ, USA, 2012), IEEE, pp. 214–221.

[LPG12] Liu Y., Prabhakaran B., Guo X.: Point-based manifold harmonics. *IEEE Transactions on Visualization and Computer Graphics 18*, 10 (2012), 1693–1703.

[LS03] Labelle F., Shewchuk J. R.: Anisotropic Voronoi diagrams and guaranteed-quality anisotropic mesh generation. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry* (New York, USA, 2003), ACM, pp. 191–200.

[LZ07] Liu R., Zhang H.: Mesh segmentation via spectral embedding and contour analysis. *Computer Graphics Forum 26* (2007), 385–394.

[MBR11] Macdonald C. B., Brandman J., Ruuth S. J.: Solving eigenvalue problems on curved surfaces using the closest point method. *Journal of Computational Physics 230*, 22 (2011), 7944–7956.

[MDSB03] Meyer M., Desbrun M., Schröder P., Barr A. H.: Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and Mathematics III*. H.-C. Hege and K. Polthier (Eds.). Springer, Berlin, Germany (2003), pp. 35–57.

[PG01] Pauly M., Gross M.: Spectral processing of point-sampled geometry. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (New York, USA, 2001), ACM, pp. 379–386.

[PGK02] Pauly M., Gross M., Kobbelt L.: Efficient simplification of point-sampled surfaces. In *Proceedings of IEEE Visualization 02* (Piscataway, NJ, USA, 2002), R. Moorhead, M. Gross and K. I. Joy (Eds.), IEEE, pp. 163–170.

[PPH*13] Petronetto F., Paiva A., Helou E., Stewart D., Nonato L.: Mesh-free discrete Laplace–Beltrami operator. *Computer Graphics Forum 32* (2013), 214–226.

[TAOZ12] Tagliasacchi A., Alhashim I., Olson M., Zhang H.: Mean curvature skeletons. *Computer Graphics Forum 31* (2012), 1735–1744.

[Tau95] Taubin G.: A signal processing approach to fair surface design. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques* (New York, USA, 1995), ACM, pp. 351–358.

[VL08] Vallet B., Lévy B.: Spectral geometry processing with manifold harmonics. *Computer Graphics Forum 27* (2008), 251–260.

[WMKG07] Wardetzky M., Mathur S., Kälberer F., Grinspun E.: Discrete laplace operators: No free lunch. In *Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, 2007), pp. 33–37.

[YZX*04] Yu Y., Zhou K., Xu D., Shi X., Bao H., Guo B., Shum H.-Y.: Mesh editing with poisson-based gradient field manipulation. *ACM Transactions on Graphics (TOG) 23* (2004), 644–651.