Intrinsic Girth Function for Shape Processing

SHI-QING XIN
Ningbo University, China
WENPING WANG
The University of Hong Kong
SHUANGMIN CHEN and JIEYU ZHAO
Ningbo University, China
and
ZHENYU SHU
Ningbo Institute of Technology, Zhejiang University, China

Shape description and feature detection are fundamental problems in computer graphics and geometric modeling. Among many existing techniques, those based on geodesic distance have proven effective in providing intrinsic and discriminative shape descriptors. In this article we introduce a new intrinsic function for a three-dimensional (3D) shape and use it for shape description and geometric feature detection. Specifically, we introduce the *intrinsic girth function* (IGF) defined on a 2D closed surface. For a point p on the surface, the value of the IGF at p is the length of the shortest nonzero geodesic path starting and ending at p. The IGF is invariant under isometry, insensitive to mesh tessellations, and robust to surface noise. We propose a fast method for computing the IGF and discuss its applications to shape retrieval and detecting tips, tubes, and plates that are constituent parts of 3D objects.

Categories and Subject Descriptors: I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling—*Geometric algorithms*; I.3.8 [**Computer Graphics**]: Applications

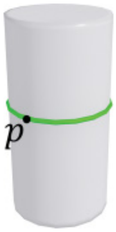
General Terms: Algorithms

Additional Key Words and Phrases: Intrinsic girth function, geodesic loop, shape signature, tip identification

ACM Reference Format:

Shi-Qing Xin, Wenping Wang, Shuangmin Chen, Jieyu Zhao, and Zhenyu Shu. 2016. Intrinsic girth function for shape processing. ACM Trans. Graph. 35, 3, Article 25 (March 2016), 14 pages.
DOI: http://dx.doi.org/10.1145/2866570

1. INTRODUCTION



We introduce a new shape function, called the *intrinsic girth function* (IGF), which is defined on the boundary surface of a three -dimensional (3D) shape, and study its applications to shape analysis and shape retrieval. Briefly speaking, the value of the IGF at a point p on a closed surface $\mathcal{M}$ is the length of the shortest nonzero geodesic path on $\mathcal{M}$ that starts and ends at p (see the right inset figure). Intuitively, it is the girth of the surface $\mathcal{M}$ based at p, hence the name *intrinsic girth function*.

Since the IGF is intrinsically defined with geodesics, it is invariant under isometry, which is a shape change that preserves geodesic distances. Therefore, the IGF is oblivious to pose changes of a 3D shape, since pose changes are typically an approximate isometry in applications such as animation and shape deformation. Furthermore, we will show that the IGF is insensitive to mesh tessellations and robust to surface noise up to a reasonable level. Due to these properties, the IGF induces an effective shape signature for shape matching. We will also show that the IGF can be used to effectively detect the constituent parts of 3D objects that are tips, tubes, and plates, respectively.

To facilitate the application of the IGF, we will present an efficient method for computing the IGF of triangle mesh surfaces. Our method is built on existing exact geodesic algorithms [Mitchell et al. 1987; Chen and Han 1990; Surazhsky et al. 2005; Xin and Wang 2009; Liu 2013] that discretize all the shortest paths into a finite number of windows, each encoding a set of shortest paths sharing a common face sequence. A naïve adaption of the existing methods would consider all the possibilities that a pair of windows meets at a mesh edge or vertex, leading to an algorithm of $O(n^4 \log n)$ time and $O(n^2)$ space, which would be too inefficient for practical applications. We observe that only windows near the circumference of an expanding geodesic disk centered at a point p may contribute to determining the value of IGF at p and hence need to be processed.

This work is supported by the NSF of China (61300168, 61272019, 61332015, 61571247, 11226328), GRF (HKU717813E), the Open Project Program of the State Key Laboratory of CAD&CG (A1412), the Opening Foundation of Zhejiang Provincial Top Key Discipline (XKXL1406 and XKXL1429), ZJNSF (LZ16F030001 and LY13F020018), and the NSF of Ningbo City (2013A610058 and 2015A610123).

Authors' addresses: S.-Q. Xin, S. Chen, and J. Zhao, Ningbo University, PR China, 315211; emails: xinshiqing@163.com, chenshuangmin@nbu.edu.cn, zhao_jieyu@nbu.edu.cn; W. Wang, Department of Computer Science, The University of Hong Kong (HKU), PR China, 999077; email: wenping@cs.hku.hk; Z. Shu, School of Information Science and Engineering, Ningbo Institute of Technology, Zhejiang University, Ningbo, PR China, 315100; email: shuzhenyu@nit.zju.edu.cn.

DOI: http://dx.doi.org/10.1145/2866570

This observation enables us to develop a set of effective speed-up techniques to develop a fast and practical algorithm.

Contributions: We present the concepts, computation, and applications of the IGF. Our main contributions are as follows:

(1) The definition of the IGF as a new shape function;
(2) An efficient algorithm for computing the IGF of a triangulated surface;
(3) A study on the application of the IGF as an isometry-invariant shape signature to shape matching and shape retrieval;
(4) A study on the application of the IGF to shape structure analysis in detecting tips, tubes, and plates that are constituent parts of 3D shapes.

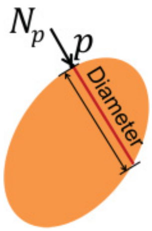
2. RELATED WORK

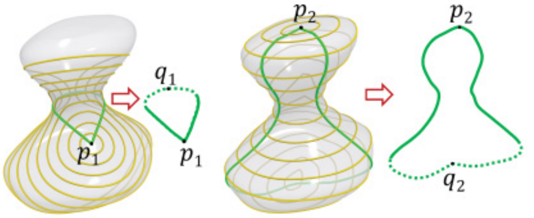
In the following we will review some representative works on shape signatures and discrete geodesics, which are related to our work.

Shape Signatures. Since large databases of 3D models are becoming available, shape retrieval [Osada et al. 2002; Gal et al. 2007; Tangelder and Veltkamp 2008; Chang et al. 2003; Zimmer et al. 2013; Litman et al. 2014] and shape analysis [Ovsjanikov et al. 2012; Sun et al. 2009] are of an increasing importance. Generally, we need to convert 3D objects of interest into a compact representation for this purpose. An effective shape signature should be invariant to scale, translation, and rotation, as is the case with Shape Context [Belongie et al. 2000] and Spin Image [Johnson and Hebert 1999; Assfalg et al. 2007]. An important class of these techniques is to develop some shape signatures able to discriminate different shapes but oblivious to different poses of the same shape [Lian et al. 2013; Gal et al. 2007; Shapira et al. 2008] or invariant under isometry [Aubry et al. 2011; Sun et al. 2009; Bronstein and Kokkinos 2010]. We refer to Osada et al. [2002], Gatzke et al. [2005], Tangelder and Veltkamp [2008], Bronstein et al. [2011], and Mendoza [2011] for a comprehensive survey on shape retrieval.

One kind of research work on shape signatures has been focusing on defining informative descriptors that are often high dimensional. For example, several spectral methods [Sun et al. 2009; Bronstein and Kokkinos 2010; Aubry et al. 2011; Litman et al. 2014] based on the Laplace-Beltrami operator associated with the shape have been proposed in recent years. The resulting descriptors are often of a high complexity and have to be mapped to another space spanned by a vocabulary of typical feature vectors before shape retrieval [Bronstein et al. 2011; Litman et al. 2014]. Another line of methods integrate multiple features to improve the shape retrieval performance [Kuang et al. 2015; Chen and Chiang 2010]. Clearly, a desirable descriptor should be able to capture both isometric properties [Ion et al. 2007, 2008] and perceptual features [Gal et al. 2007] in order to improve the overall discriminative ability. This motivates us to propose a new descriptor, called the IGF.

Shape Diameter Function. We are going to review in more detail the *shape diameter function* (SDF) introduced by Gal et al. [2007] because of its similarity to the IGF proposed in the present article and because we will use the SDF in junction with the IGF for shape retrieval and part analysis of 3D shapes. The SDF is defined on the boundary surface of a 3D object that measures the local thickness of the object. The SDF is similar to the medial axis transform [Choi et al. 1997] but it circumvents the notorious difficulty of robust computation of the medial axis transform. Consider a point p on a closed smooth surface $\mathcal{M}$. Let N_p be the ray that starts at p and goes in the direction of the inward normal vector of the surface $\mathcal{M}$ at p. The value of the SDF at p, called the *shape diameter*, is the distance from the point p to the first intersection of the ray N_p with the surface $\mathcal{M}$; See the inset figure. An approximate method [Gal et al. 2007] is proposed for computing the SDF of a mesh surface.





(a) Tangential contact (b) Contracting into a point

Fig. 1. Expanding geodesic disks. (a) The frontier curve comes into tangential contact with itself at a point such that the disk becomes a region that is no longer singly connected. (b) The entire frontier curve contracts simultaneously into a single point.

Gal et al. [2007] apply the SDF to define a pose-oblivious shape signature. Shapira et al. [2008] and Fan et al. [2011] use the SDF for mesh partitioning and skeletonisation. Shapira et al. [2010] use the SDF to find analogies between parts of 3D objects. Note that the SDF is not invariant under isometry, since its definition depends the 3D embedding of a 3D shape rather than based on geodesic properties. Nevertheless, it is used quite successfully for shape matching with pose changes, since local thickness normally does not change much with pose changes in applications. In contrast, the IGF introduced in the present article is invariant under isometry. Our tests show the IGF is better than the SDF for shape retrieval. Furthermore, the combination of the IGF and the SDF yields a very competitive shape signature in comparison with other prevailing shape signatures.

Discrete Geodesics. By the term "discrete geodesic," we mean a polygonal path on a mesh surface $\mathcal{M}$ that is locally shortest everywhere. Sharir and Schorr [1986] propose the first exact algorithm for computing discrete geodesics on convex polyhedra with an $O(n^3 \log n)$ time complexity. Mitchell *et al.* (MMP) [1987] improve the time complexity bound to $O(n^2 \log n)$ using the "continuous Dijkstra" technique. Chen and Han (CH) [1990] build a binary tree to encode all the edge sequences that can possibly contain a shortest path, thereby improving the time complexity to $O(n^2)$. Some variants of these exact algorithms, such as the algorithms in Surazhsky et al. [2005] and Liu [2013], are based on MMP's algorithm, while the others in Xin and Wang [2009], Ying et al. [2014], and Ying et al. [2013] are based on CH's algorithm. Approximation methods are often used for computing geodesics in computer graphics, with the fast marching method [Kimmel and Sethian 1998] being the most popular. Crane et al. [2013] propose an approximation algorithm based on heat diffusion. Solomon et al. [2014] show that the earth mover's distance (EMD) between two delta distributions reduces to the geodesic distance.

3. INTRINSIC GIRTH FUNCTION

3.1 Definition

Now we are going to give the definition of the IGF and discuss its properties. Consider the expanding disk that is centered at the point p and expands on the surface $\mathcal{M}$, as shown in Figure 1. The circumference of this expanding disk will also be referred to as the *frontier curve*, which is an isovalue curve of the geodesic distance field centered at p. When the disk grows sufficiently large on $\mathcal{M}$, the frontier curve will become self-intersecting.

There are two cases where this self-intersection can occur, leading to two cases for defining the IGF. The generic case of self-intersection is illustrated in Figure 1(a), where the frontier curve comes into tangential contact with itself at a point, denoted q, such that the disk becomes a region that is no longer singly connected. Evidently, there are two distinct geodesic paths from q to p which, when joined together, form a p-based geodesic loop $\mathring{\gamma}_p$, because they have the parallel tangent directions at the point q. This follows from the property that a geodesic starting at a point is uniquely determined by its starting direction. This p-based geodesic loop $\mathring{\gamma}_p$ is the globally shortest p-based geodesic loop, since we use the first point of self contact of the expanding disk. Clearly, the value of the IGF at the point p, which is defined as the length of $\mathring{\gamma}_p$, equals twice the radius of the geodesic disk when the contact at q is formed.

The shortest p-based geodesic loop may also arise in the second, special case where the expanding geodesic disk centered at p forms a self-intersection in a manner that differs from the preceding generic case, as shown in Figure 1(b). Here the entire frontier curve of the disk contracts simultaneously into a single point q. It can be shown that there are at least three geodesics paths of the same length from q to p; see Sharir and Schorr [1986] for a detailed proof. Hence, in this case, the value of the IGF at p is taken to be twice the radius of the disk.

The difference between this special case and the generic case is as follows. Let $F_p(x)$ denote the geodesic distance function defined by the geodesic distance from a point $x \in \mathcal{M}$ to the source point p. In the generic case, the tangential contact point q is a saddle point of $F_p(x)$. Note that when $\mathcal{M}$ is of genus $g \geq 1$, $F_p(x)$ has at least one saddle point and the contact point q always exists for any $p \in \mathcal{M}$. In the special case, however, $F_p(x)$ has no saddle points and the contracting point q is the unique local maximum point of $F_p(x)$. It happens only if $\mathcal{M}$ is of genus 0.

3.2 Relation of IGF to Injectivity Radius

To avoid possible confusion, we need to point out that the intrinsic girth function is closely related but not equivalent to the *injectivity radius* associated with the exponential map at a point of a surface [Abresch and Meyer 1997]. Given a 2D smooth surface $\mathcal{M}$, consider the exponential map $\exp_p$ at a point $p \in \mathcal{M}$: $T_p(\mathcal{M}) \to \mathcal{M}$, where $T_p(\mathcal{M})$ is the tangent space of $\mathcal{M}$ at p. Let $\mathcal{B}(p, r) \subset T_p(\mathcal{M})$ be a disk in $T_p(\mathcal{M})$ that is centered at p and has radius $r > 0$. Then the exponential map maps $\mathcal{B}(p, r)$ to the geodesic disk $\mathcal{D}(p, r) \subset \mathcal{M}$ centered at p and having radius r. When r is sufficiently small, the exponential map from $\mathcal{B}(p, r)$ to $\mathcal{D}(p, r)$ is a differeomorphism (i.e., it is injective and its inverse is also differentiable) and there is a unique minimizing geodesic connecting p to any point in $\mathcal{D}(p, r)$.

However, when r is sufficiently large, the exponential map from $\mathcal{B}(p, r)$ to $\mathcal{D}(p, r)$ will stop being differeomorphism. Let $\exp_{(p,r)}$ denote the restriction of $\exp_p$ to $\mathcal{B}(p, r)$, which maps $\mathcal{B}(p, r)$ to $\mathcal{D}(p, r)$. The *injectivity radius* at p is defined to the maximum possible value of r such that the restricted exponential map $\exp_{(p,r)}$ is a differeomorphism. It is known that $\exp_{(p,r)}$ fails to be a differeomorphism in only two ways [James 1962].

(1) *Case* 1: As shown in the top row of Figure 2 (also seen in Figure 1(a)), the boundary of $\mathcal{D}(p, r_0)$ forms a self-intersecting point, denoted q, such that there are two distinct minimizing geodesics from p to q, and these two paths have the same length r_0. In this case the mapping $\exp_{(p,r)}$ is no longer injective and therefore not a differeomorphism.

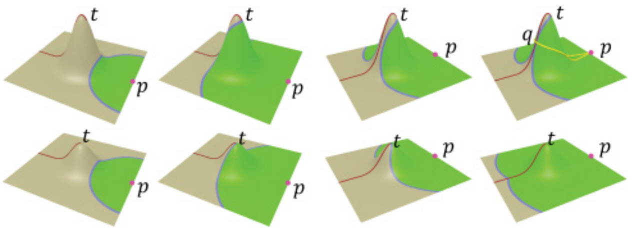


Fig. 2. Conjugate points and ridge points. The top row shows how a geodesic disk centered at the point p goes around a "peak" on a surface. Here the frontier curve of the disk form a self-intersection to yield a p-based shortest geodesic loop, shown as the yellow curve, whose length defines the value of the IGF at p. In fact, q is the first ridge point touched by the frontier curve of the expanding geodesic disk. Furthermore, the two geodesic curves from p to q have parallel tangent directions at the point q. When the "peak" is lowered to become a relatively flat bump, as shown in the bottom row, the frontier curve of the geodesic disk first develops a conjugate point, denoted t, at which the exponential map at p has a vanishing Jacobian. Beyond the conjugate point a nonsmooth point is developed on the boundary of the disk.

(2) *Case* 2: The boundary of $\mathcal{D}(p, r_0)$ contains a point, denoted t, that is conjugate to p along the geodesic from p to t [Hopf 1948; Polthier and Schmies 1999]. As shown in the bottom row of Figure 2, this case occurs, for example, when the surface bump is not as pronounced as the case of the top row of Figure 2.

In the first case, the mapping $\exp_{(p,r_0)}$ has a vanishing Jabobian at q. In this case, $\exp_{(p,r_0)}$ is still injective except at q but its inverse is not differentiable. Hence $\exp_{(p,r_0)}$ not a differeomorphism. The two distinct minimizing geodesics from p to q, when joined together at q, form the shortest geodesic path that starts and ends at p, which, by definition, is the p-based geodesic loop at p. Therefore, the value of the IGF at p is $2r_0$.

In the second case, because conjugate points can easily be caused by local, small-scale surface variations or bumps, we do not use the injectivity radius r_0 to define the IGF; instead, we let the geodesic disk $\mathcal{D}(p, r)$ continue to grow beyond the injectivity radius r_0 to reach some radius $r_1 > r_0$ such that the first self-intersecting point is formed on the boundary of $\mathcal{D}(p, r_1)$, and we define the value of the IGF at p to be $2r_1$. In this way, by ignoring the injectivity radius arising due to conjugate points, we make the IGF immune to small scale surface variations or surface noise to provide a robust shape description.

To summarize, the IGF is related but not equivalent to the injectivity radius. The IGF is immune to local, small surface variations, while the injectivity radius is not. Furthermore, the value of the IGF at any point p is equal to or greater than twice the injectivity radius.

4. COMPUTATION OF INTRINSIC GIRTH FUNCTION

4.1 Preliminaries

We first introduce necessary notations and basic data structures to be used in our method.

$\mathcal{M}$: the input polygonal mesh;
e, e^-: a mesh half-edge and its negate;
w: a window that encodes a set of shortest paths sharing the same edge sequence;

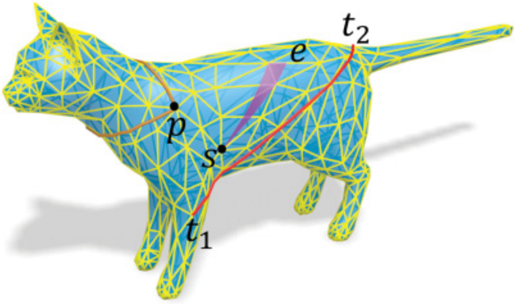


Fig. 3. Illustration of discrete geodesics, p-based geodesic loops and windows. The red curve is the shortest path between the two points t_1 and t_2. The orange curve is the p-based geodesic loop. The purple region is a window that encodes a set of shortest paths starting from the point s.

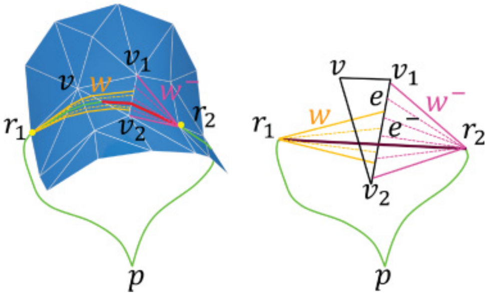
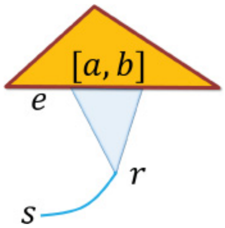


Fig. 4. The right figure shows a flattening of the surface on the left. Note that the windows $w = (r_1, l_1, e, [a_1, b_1])$ and $w^- = (r_2, l_2, e^-, [a_2, b_2])$ determine a p-based geodesic loop if and only if the line segment $\overline{r_1 r_2}$ goes through the interval $[a_1, b_1] \cap [a_2, b_2]$, where $[a_1, b_1]$ and $[a_2, b_2]$ are respectively the ending intervals of w and w^- on the edge e.

$\mathcal{Q}$: a priority queue of windows;
R_g: the propagation radius of wavefronts;
$\gamma(p, q)$, $\ell(p, q)$: the shortest path between p and q, and the corresponding length;
$\mathring{\gamma}_p$: the p-based geodesic loop, that is, a closed geodesic curve on $\mathcal{M}$ starting from and ending at p;
$\mathcal{D}(p, r)$: the geodesic disk rooted at p with the radius r;
$I(p)$: the value of the IGF at the point $p \in \mathcal{M}$.

As shown in Figure 3, the shortest paths from the point s to the points in the purple region share the same edge sequence. We use a quadruple $(r, l, e, [a, b])$, called a *window* w, to encode the set of shortest paths, where r is the unfolded image of the last vertex of the vertex-edge sequence, l is the geodesic distance between s and r, e is the mesh edge where w arrives, and $[a, b]$ is the segment covered by w (see the inset on the right figure). Windows are further classified into *pseudosource windows* that arrive at mesh vertices and *interval windows* that arrive at mesh edges. Windows are a commonly used data structure used in existing methods for exact computation of discrete geodesics [Mitchell et al. 1987; Surazhsky et al. 2005; Chen and Han 1990; Xin and Wang 2009]. These exact algorithms differ mainly in the way they avoid the combinatorial explosion of the number of windows[1]. All these exact algorithms can be adapted to computing the IGF. We use the ICH algorithm in this article.



4.2 A Naïve Algorithm

To compute the IGF at p, we need to compute the shortest p-based geodesic loop $\mathring{\gamma}_p$. This is done by finding a pair of windows at the frontier curve of the expanding geodesic disk that meets each other in opposite directions. In the following we will first present a naïve algorithm for this task. This simple algorithm is not intended for practical use but helps the reader understand the optimized method we will present later in Section 4.3.

A straightforward algorithm for finding a p-based geodesic loop candidate works as follows:

Step 1: Compute the exact geodesic distance field rooted at the given point p and keep all the windows generated;

Step 2: For each mesh edge $\overline{vv'}$, check if $\gamma(p, v) \cup \gamma(v, v') \cup \gamma(v', p)$ is locally shortest at v and v' [Mitchell et al. 1987] and take the composite path as a p-based geodesic loop candidate if it is locally shortest at v and v' at the same time.

Step 3: Refer to Figure 4. For each mesh edge e, we check each pair of windows w associated with e and w^- associated with e^- to see if they are able to determine a p-based geodesic loop. Let $w = (r_1, l_1, e, [a_1, b_1])$ and $w^- = (r_2, l_2, e^-, [a_2, b_2])$. Then w, together with w^-, determines a p-based geodesic loop if and only if the straight line connected by r_1 and r_2 goes through $[a_1, b_1] \cap [a_2, b_2]$.

By its definition, we can determine the value of the IGF at a point p, denoted $I(p)$, by finding a loop candidate with the minimum length. The IGF of the whole surface could then be computed if the algorithm is applied to every vertex $v \in \mathcal{M}$. This method is inefficient in both time and space because: (1) it requires $O(n^2)$ space to maintain the windows for a mesh with n vertices and (2) the worst-case time complexity of ICH is $O(n^2 \log n)$ and every newly generated window has to be matched with the other $O(n)$ existing windows on the same mesh edge (the number of windows on a mesh edge is $O(n)$ [Chen and Han 1990]), the naïve algorithm needs $O(n^3 \log n)$ time to compute the shortest p-based geodesic loop and thus the overall time complexity is $O(n^4 \log n)$.

4.3 Speed-Up Techniques

The above naïve algorithm is inefficient because it maintains and processes many loop candidates that clearly do not contribute to the final p-based geodesic loops. We now present some speed-up techniques to devise a more efficient method. Our key idea is to process only the windows near wavefronts and terminate the algorithm as soon as the shortest p-based geodesic loop has been found. There are two aspects to consider: (1) when to terminate the algorithm and (2) whether the window w is close enough to the wavefront and therefore likely to contribute to defining the IGF.

4.3.1 *Termination Condition.*

We employ the priority queue $\mathcal{Q}$ of windows in the ICH algorithm [Xin and Wang 2009] to ensure the from-near-to-far property of wavefront propagation. The priority of the window w is measured by its minimum geodesic distance to the source point, that is, the length of the shortest one among the geodesic paths encoded by w. The propagation works as follows:

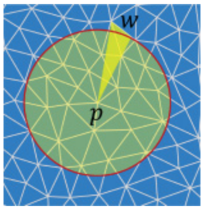
---

[1] The implementations of the MMP and ICH algorithms are available at http://research.microsoft.com/en-us/um/people/hoppe/proj/geodesics/ and https://sites.google.com/site/xinshiqing/.

ALGORITHM 1: Wavefront Propagation Scheme

```
while $\mathcal{Q}$ is not empty do
    Pop out the top-priority window $w$ in $\mathcal{Q}$;
    Update the propagation radius $R_g$ with $w$'s minimum geodesic distance;
    Compute $w$'s children and push them into $\mathcal{Q}$;
end
```

On a polygonal mesh, windows in the priority queue are ranked by their minimum geodesic distances, and R_g has to experience a step increase throughout the wavefront propagation, in line with the jump of priorities of windows; see the inset. The "from-near-to-far" propagation implies that for all the points (including face interior points) located inside the geodesic disk with the radius R_g, their geodesic distances are already determined [Mitchell et al. 1987]. In other words, all the p-based geodesic loops that are not longer than $2R_g$ have been found (i.e., can be reconstructed from the generated windows).



However, as Figure 4 shows, each p-based geodesic loop candidate has to be determined by two windows that meet at some edge, and the priority of the loop is actually up to one of the windows that comes later rather than the real length of the loop. Therefore, the shortest p-based geodesic loop cannot be guaranteed to be found earlier than other p-based geodesic loop candidates. Fortunately, the above observation gives a suitable termination condition: When the geodesic disk has a radius R_g, all the p-based geodesic loops at most $2R_g$ have been found, and thus the globally shortest p-based geodesic loop, therefore $I(p)$, has been determined when R_g is larger than half of the length of the best-so-far p-based geodesic loop $\mathring{\gamma}_p$.

4.3.2 *Identifying Windows at Disk Boundary.* The second speed-up technique is to capture the moment when the boundary of the current geodesic disk forms a self-intersecting point. For this purpose, we will check if both the windows of interest are at the disk boundary and consider specifically how they constitute a p-based geodesic loop. By the properties of discrete geodesics (see Section 3), the shortest p-based geodesic loop $\mathring{\gamma}_p$ on the mesh $\mathcal{M}$ is a polyline, each segment passing through a polygonal face. There always exists a point $q \in \mathring{\gamma}_p$ such that q has the same distance to p in two opposite directions. Let L_q be the line segment that contains q. There are three cases regarding the location of q, as shown in Figure 5:

Case 1: L_q is a mesh edge, denoted $\overline{vv'}$ (see Figure 5 (top-left));

Case 2: One endpoint of L_q is the mesh vertex v, and the other is a point on the edge e; q is closer to v (see Figure 5 (top-right));

Case 3: At least one endpoint of L_q is not a vertex and the endpoint closer to q is not a vertex (see Figure 5 (bottom)).

Each time when a new window is created, we check if it is likely to contribute to the determination of $I(p)$. If the newly created window w is a pseudosource window (i.e., reaching a mesh vertex), then we consider Case 1 and Case 2. If w is an interval window (i.e., reaching a mesh edge), then we consider Case 2 and Case 3. Actually, Case 2 can be absorbed into Case 3. The reason is as following. Case 2 describes the situation that a pseudosource window w_1 and an interval window w_2 determine a p-based geodesic loop together, and w_1 comes later than w_2. Considering that w_1 will immediately produce an interval child, say, w_1', on the mesh edge of w_2's interval, it is not late to consider the same loop using w_1' and w_2. Note that this convenience in implementation is at the cost of the tight checking condition; see the first requirement of Lemma 4.2.

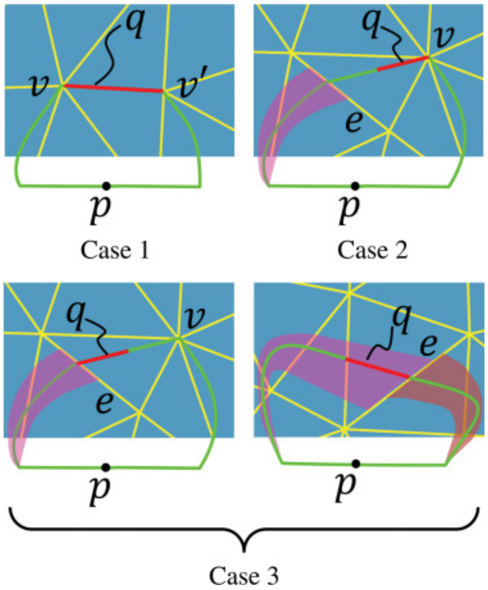


Fig. 5. Let q be the farthest point constrained on the p-based geodesic loop $\mathring{\gamma}_p$, that is, there are two equal-in-length shortest paths between p and q. The location of q can be in three cases, where the segment in red shows the possible range of q.

The key points in checking if w is at disk boundary include:

(1) Determine if w is **close** enough to the farthest point q.
(2) Determine if w can give a closed curve that is locally shortest around q.
(3) Determine if the loop given by w is the best-so-far candidate loop.

The following lemmas provide the rules for checking if pseudosource windows and interval windows are at the disk boundary. The proofs are given in the Appendix.

LEMMA 4.1. *Suppose the window w popped out from the priority queue is a pseudosource window rooted at v. Let v' be a one-ring neighbor vertex of v. The situation that the loop $\mathring{\gamma}_p = \gamma(p, v) \cup \overline{vv'} \cup \gamma(v', p)$ defines $I(p)$ belongs to Case 1 if the following conditions are met:*

(1) $\ell(p, v') \leq \ell(p, v)$;
(2) $\mathring{\gamma}_p$ *is locally shortest at* v *and* v'*;*
(3) $\|\mathring{\gamma}_p\|$ *is shorter than the best-so-far* p*-based geodesic loop.*

LEMMA 4.2. *Suppose the window w popped out from the priority queue is an interval window arriving at the edge $e = \overline{v_1 v_2}$, and w^- is an interval window on the reverse edge e^-. The situation that the windows w and w^- define $I(p)$ belongs to Case 2 or Case 3 if*

(1) $\ell_{max}(w^-) + 2\max(\|v_1 v\|, \|v_2 v\|) \geq \ell_{min}(w)$, *where* v *is the vertex opposite to* e*, and* ℓ_{max} *and* ℓ_{min} *respectively denote the maximum and minimum geodesic distances given by the corresponding interval window;*
(2) *The straight line by connecting* w *and* w^-*'s roots goes through* w *and* w^-*'s common interval;*

(*3*) *The new loop $\mathring{\gamma}_p$ is shorter than the best-so-far p-based geodesic loop.*

Remark: In Lemma 4.2, if $\ell_{\max}(w^-) + 2\max(\|v_1 v\|, \|v_2 v\|) < \ell_{min}(w) = R_g$, then w^- is far behind the wavefront (i.e., inside the p-based geodesic disk) and no longer useful for determining the shortest p-based geodesic loop. Hence we delete w^- from the window list on the edge e^-in this case. This is helpful in performance boosting and memory reduction.

4.4 The Complete Algorithm

Combining the aforementioned termination condition and checking rules yields an efficient algorithm for computing the IGF, as shown in Algorithm 2. Since it is difficult to establish a theoretically tight bound on the time and space complexities of the algorithm, we will provide experimental validations in Section 5 to demonstrate its improved efficiency.

ALGORITHM 2: Computing $I(p)$

```
Input: A closed polygonal mesh $\mathcal{M}$; A point $p \in \mathcal{M}$; A
       priority queue $\mathcal{Q}$ of windows;
Output: The length of the shortest $p$-based geodesic loop,
        $I(p)$;
while $\mathcal{Q}$ is not empty do
    Pop out the first window $w$ in $\mathcal{Q}$;
    if $w$ is a pseudo-source window at $v$, then
        Check $v$ and its one-ring neighboring vertices using
        Lemma 4.1 and update the best-so-far $\mathring{\gamma}_p$ if
        necessary;
    else
        /*$w$ is an interval window, say, $w \in e$.*/
        Check $w$ and each window $w^- \in e^-$ using Lemma
        4.2;
        if the first condition of Lemma 4.2 doesn't hold, then
            Delete $w^-$ from $e^-$.
            if $w$ and $w^-$ satisfy the second and last
            conditions of Lemma 4.2, then
                Update $\mathring{\gamma}_p$ by $w$ and $w^-$.
            end
        end
    end
    Compute $w$'s children and push them into $\mathcal{Q}$.
end
if the shortest $p$-based geodesic loop $\mathring{\gamma}_p$ is found, then
    Return $\|\mathring{\gamma}_p\|$;
else
    Return $2\ell(p, q)$, where $q$ is the farthest point from $p$.
end
```

5. VALIDATION

In this section we use extensive experimental results to validate our method for computing the IGF in terms of time and space efficiency, as well as its insensitivity to mesh quality and its robustness to small surface variations and topological changes.

5.1 Performance

We implemented and experimented with the IGF algorithm on a computer with a 64-bit version of Win7 system, a 3.07GHz Intel(R) Core(TM) i7 CPU and 6GB memory. The coding language is C++ supported by OpenMP. We first give performance statistics with and without the speed-up techniques in Table I. Then, in Figure 6, we show the timing plots for six models of resolutions ranging from 2K to 20K faces.

Table I. Performance Statistics with and without the Speedup Techniques. The Test Was Made on the Horse Model Shown in Figure 6

	Naïve		Improved	
#Faces	Timing (s)	Memory (Mb)	Timing (s)	Memory (Mb)
2K	8.0	3.0	0.25	0.2
4K	55.6	10.3	0.76	0.4
6K	173.3	22.0	2.14	0.7
8K	395.5	38.8	4.39	1.2
10K	767.6	61.3	7.73	1.5

Even with the speed-up techniques proposed in Section 4, our IGF algorithm is still quite time-consuming when applied to objects with a large number of mesh vertices. For instance, the processing time is about 506.9s for the Elephant mesh (Figure 7) with 50K faces. Next we will show that the IGF is insensitive to mesh resolution. Thus, for better efficiency, one may compute the IGF on a reasonably simplified mesh model of an input object.

To investigate the influence of mesh resolution to the IGF, we compare the cumulative distribution functions (CDF) of the normalized IGFs on five Elephant models of different levels of simplification, normalized by the total area. See Figure 7(f). The difference between two IGFs is computed by

$$d = \int_{I_{min}}^{I_{\max}} |\mathrm{CDF}_1 - \mathrm{CDF}_2| \mathrm{d}I, \tag{1}$$

where I is the IGF variable and $I_{\max}$ and $I_{\min}$ are respectively the maximum and minimum IGFs. Visually, the IGFs computed at the different levels of simplification are all very close to that of the original model, with the maximum difference being 8.7×10^{-3}, while the computation time cost is dramatically reduced on the simplified versions. For instance, it takes 16.2s for 10K faces and 0.5s for 2K faces. Because of the insensitivity of the IGF to mesh resolution, all the models shown in this paper are remeshed to have 10K faces when we apply the IGF for shape retrieval later in Section 6.1.

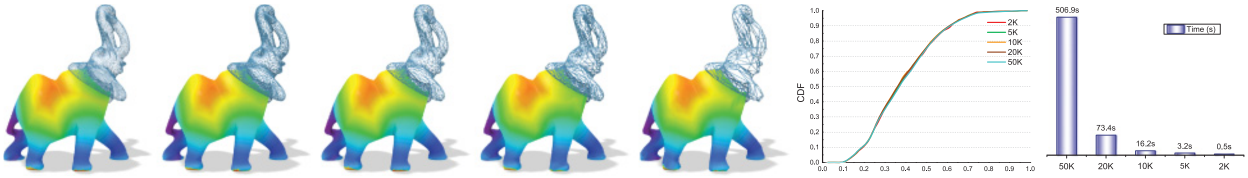
5.2 Accuracy and Robustness

When different discrete meshes are derived from the same underlying smooth model, the accuracy of our IGF method is insensitive to the meshing quality, because exact geodesic paths pass through face interior points, rather than being constrained to mesh edges.

Figure 8 shows six versions of the Fertility model with differences in mesh density and quality, as well as geometric and topological variations. Still using Equation (1) to estimate the differences between IGFs, the errors of the variants (see Figures 8(b)–(f)) relative to the original model (see Figure 8(a)) are quite small: (b) 7.7×10^{-3}, (c) 5.3×10^{-3}, (d) 1.2×10^{-2}, (e) 2.7×10^{-2}, and (f) 5.9×10^{-2}. Hence, we conclude that the IGF is insensitive to mesh tessellations and resistant to mild geometric noise (see Figure 8(d)) and flat humps (see Figure 8(e)). Even when the input model has some topological changes (see Figure 8(f)), the IGF still gives a meaningful result: The majority of the IGF values remain unchanged.

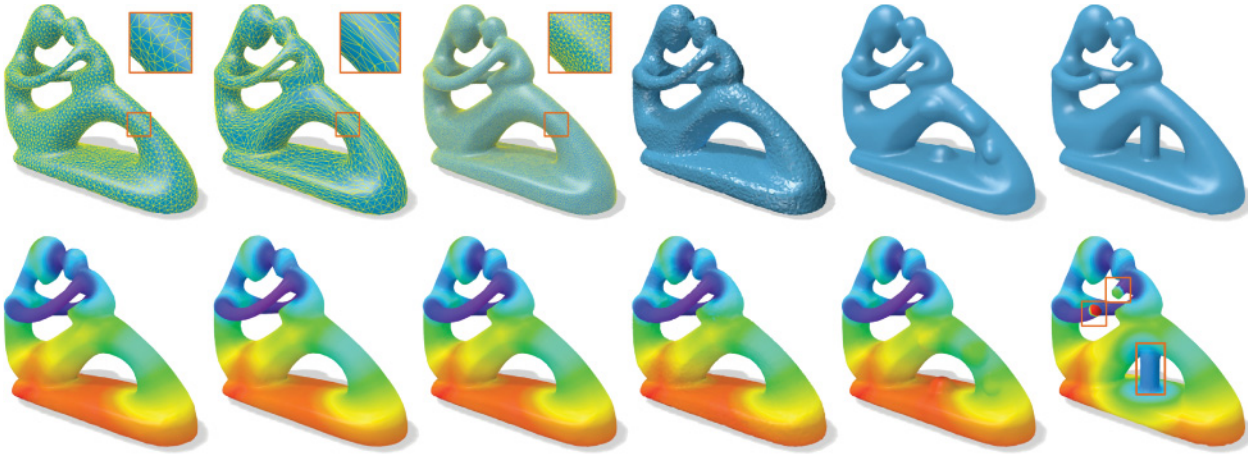
Figure 9 shows that the IGF is pose oblivious. Due to the nice geometric properties, the IGF can be used to define an effective shape aware signature, as we will discuss in Section 6.1.

Fig. 6. Performance plot. Each model has different resolutions from 2K to 20K faces.



(a) 50K (b) 20K (c) 10K (d) 5K (e) 2K (f) CDFs (g) Timing

Fig. 7. The IGFs of the five Elephant models with different levels of simplification are very close to each other, whereas the computation time is dramatically reduced for the simplified versions.



(a) 25K (b) 10K (c) 50K (d) 25K (e) 25K (f) 25K

Fig. 8. The IGF is insensitive to differences in mesh surface quality. (a) An isotropic mesh; (b) an anisotropic mesh; (c) a denser isotropic mesh; (d) a mesh with surface noise; (e) a mesh with relatively flat humps; (f) the majority of the IGF values remains unchanged even with two changes in the topology of this model.

6. APPLICATIONS

In this section, we discuss two applications of the IGF. One is to use the IGF as an isometry-invariant shape signature to shape matching and shape retrieval, and the other is to detect constituent parts of 3D shapes, including tips, tubes, and plates.

6.1 Shape Retrieval

As mentioned in Section 1, the girth of the point p is related to the nearest saddle point of the p-based geodesic distance field. This observation naturally leads to an approximate implementation that runs the Fast Marching Method [Kimmel and Sethian 1998] and finds the saddle point of the Morse-Smale complex that is nearest to p [Huang et al. 2008]. This approximate method will be referred to as *IGF-MS*. Beside IGF and IGF-MS, the other shape signatures used for comparison include:

(1) The shape diameter function (SDF) [Shapira et al. 2008];
(2) The scale-invariant heat kernel signature (SIHKS) [Bronstein and Kokkinos 2010];

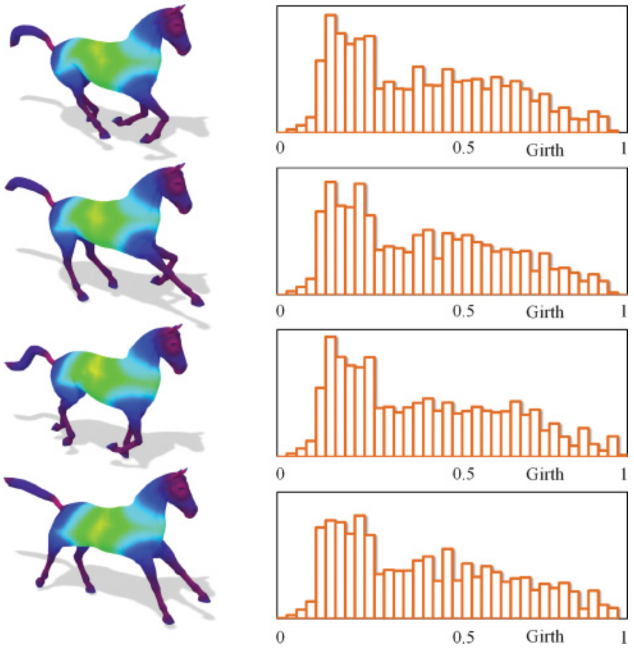


Fig. 9. The histograms of the IGFs for the horse model with different poses demonstrate the pose invariance of the IGF. The horizontal axis stands for scaled girth values, with the maximum girth being scaled to 1.

(3) The wave kernel signature (WKS) [Aubry et al. 2011];
(4) The average geodesic distance function (GeoAverage, also called the *geodesic centricity function*) [Ion et al. 2007; Martinek et al. 2012; Shamir et al. 2004];
(5) The farthest geodesic distance function (GeoFarthest, also called the *geodesic eccentricity function*) [Martinek et al. 2012].

In order to compute and compare these shape signatures, an input object is first normalized such that its total surface area equals 1 and the model is remeshed to have 10K faces. In the discrete setting, a shape signature can be represented by a scalar or vector sequence, weighted by the area of influence of each vertex, $w_1, w_2, \ldots$. A standard measure for characterizing the similarity between two signatures is the earth mover's distance [Pele and Werman 2009; Rubner et al. 2000] (EMD), also known as the Wasserstein metric.

Let $(\xi_1, \xi_2, \ldots, \xi_m)$ and $(\eta_1, \eta_2, \ldots, \eta_n)$ be the signatures of two separate models, where ξ_i and η_j are k-dimensional vectors. The EMD is to minimize the following objective function:

$$\mathcal{D}(\xi, \eta) = \min_{f_{ij}} f_{ij} d_{ij},$$

where d_{ij} is the distance between ξ_i and η_j and f_{ij} satisfies

$$\sum_j f_{ij} = w_i^{\xi}, \sum_i f_{ij} = w_j^{\eta}.$$

Note that here we take "=" rather than "≤" in the constraints since $\sum_i w_i^{\xi} = \sum_j w_j^{\eta} = 1$ holds for normalized surfaces. Specifically, the EMD is equivalent to the difference between their CDFs when ξ_i and η_j are 1D. However, when ξ_i and η_j are high dimensional, for example, SIHKS and WKS, computing the EMD is both time and space inefficient even if we use the optimized implementation [Pele and Werman 2009]. Because of this, we evaluate the retrieval performances of SIHKS and WKS based on bag of features (BoF) as in Bronstein et al. [2011], which is a histogram representation with regard to independent features.

We experimented with shape signatures on the McGill 3D Shape Benchmark [Siddiqi et al. 2008] and the SHREC'11 dataset [Lian et al. 2011], with the specific parameter values recommended in the original articles.

- Since some models in the datasets are of poor triangulation quality, leading to failure of computing eigenvalues of Laplace-Beltrami matrices, we remesh these models to have 10K faces.
- We weigh the signatures by the areas of the Voronoi regions of mesh vertices. The number of bins is set to be 512 for 1D histograms. The vocabulary size of BoF is fixed to 48.
- For the SDF [Shapira et al. 2008], the opening angle is set to be 120°, and we use 30 rays for each vertex.
- For the SIHKS [Bronstein and Kokkinos 2010], the log-scale space basis is set to be 2.0, the time scale is between 1 and 20, and the number of eigenfunctions is 19.
- For the WKS [Aubry et al. 2011], the variance of the WKS Gaussian with respect to the difference of the two first eigenvalues is set to be 6.0, and the number of evaluations of WKS is 100.
- When computing average/farthest geodesic distances and approximate IGFs, we use the Fast Marching Method [Kimmel and Sethian 1998] as the core routine.
- The commonly used measures for evaluating the retrieval performance of a descriptor include [Shilane et al. 2004; Tangelder and Veltkamp 2008; Osada et al. 2002]:
  - **Nearest neighbor (NN):** the percentage of the closest matches belonging to the same class as the query.
  - **First-tier and second-tier:** the percentage of models in the query's class that appear within the top $K - 1$ and $2(K - 1)$ matches respectively, where K is the size of the query's class.
  - **E-measure:** a composite measure of the precision and recall for a fixed number (32) of retrieved results.
  - **Discounted Cumulative Gain (DCG):** a statistic that weighs correct results near the front of the ranked list more than correct results toward the end of the list [Järvelin and Kekäläinen 2000].

The statistics of retrieval performance of these shape descriptors on the McGill Benchmark and the SHREC'11 dataset are given in Table II and Table III, respectively. Figure 10 shows the precision-recall curves, describing the relationship between precision and recall in a ranked list of matches, where the curves that are shifted up represent superior retrieval results. A group of retrieval examples for articulated models are shown in Figure 11.

Since the IGF captures geometric features in a more balanced manner, neither too global nor too local, experimental results show that the IGF outperforms all the other 1D descriptors. The IGF-MS has a slightly lower retrieval performance but runs faster than our exact geodesic algorithm due to its use of the Fast Marching Method. The GeoAverage and the GeoFarthest, in spite of their invariance to isometry, encode the geometry only in a global manner.

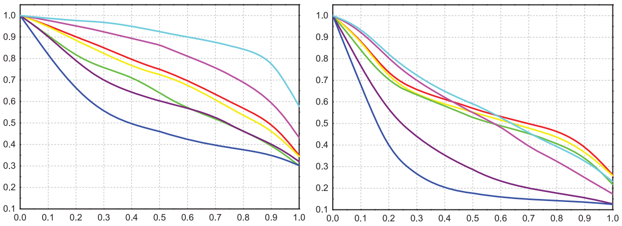
Clearly, the SIHKS and the WKS are more informative and therefore have stronger discrimination ability that the IGF, which can be explained by their high-dimensional representations. In this spirit, we tested with a composite shape descriptor that combines IGF and SDF, denoted as *IGF-SDF*, and also observed improved retrieval performance, as shown by the PR curve marked with stars in Figure 10(c) and detailed performance measures in the last row of Table III.

Table II. Retrieval Performance Comparison for the McGill 3D Shape Benchmark

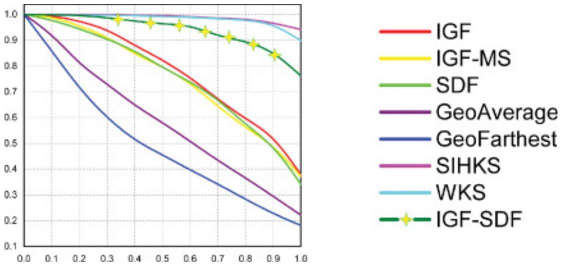
Articulated	NN	Tier 1	Tier 2	E-Measure	DCG	Non-Articulated	NN	Tier 1	Tier 2	E-Measure	DCG
IGF	93.33%	65.35%	79.56%	61.27%	90.32%	IGF	80.69%	51.47%	70.27%	50.94%	82.42%
IGF-MS	92.55%	63.40%	77.75%	59.65%	89.37%	IGF-MS	80.20%	49.93%	68.67%	49.59%	81.55%
SDF	87.45%	54.12%	74.75%	43.04%	86.31%	SDF	71.78%	47.80%	70.06%	47.46%	80.40%
GeoAverage	86.27%	53.31%	75.96%	52.96%	85.35%	GeoAverage	58.91%	30.63%	43.69%	30.14%	70.08%
GeoFarthest	72.54%	41.44%	70.88%	43.78%	79.64%	GeoFarthest	40.59%	21.51%	33.28%	21.78%	63.71%
SIHKS	97.64%	72.96%	85.65%	68.66%	93.87%	SIHKS	87.12%	46.94%	61.44%	44.91%	82.12%
WKS	98.03%	83.70%	93.83%	78.25%	96.83%	WKS	90.09%	50.16%	69.77%	49.98%	84.02%

Table III. Retrieval Performance Comparison for SHREC'11

	NN	Tier 1	Tier 2	E-Measure	DCG
IGF	99.17%	70.66%	81.13%	60.32%	92.53%
IGF-MS	98.67%	68.55%	79.28%	58.72%	91.56%
SDF	96.67%	68.40%	80.29%	58.75%	91.33%
GeoAverage	86.33%	50.98%	66.18%	47.33%	83.11%
GeoFarthest	76.50%	43.38%	56.91%	40.28%	77.90%
SIHKS	100%	97.52%	99.51%	75.94%	99.69%
WKS	99.83%	97.35%	98.85%	75.60%	99.49%
IGF-SDF	**99.83%**	**90.44%**	**95.15%**	**72.01%**	**98.03%**



(a) Articulated (McGill) (b) Non-articulated (McGill)



(c) SHREC'11; (d) Legend

Fig. 10. Precision-recall curves.

6.2 Tip Identification

We now discuss the application of the IGF to shape feature analysis–*identification of tip parts*. While there is no rigorous definition for it, a tip part of a 3D object is understood to be the end region, or the extremity, of a tubular part of a 3D object. Although tip parts are important model features needed for feature-based geometric processing, such as mesh simplification, there is a lack of effective means for their reliable identification.

Before applying the IGF to tip identification, we need to investigate the continuity of the IGF in more detail. The IGF, as well as the shortest p-based geodesic loop $\mathring{\gamma}_p$, is continuous at most points on the boundary surface of a 3D object; that is, a small change of the base point p causes only a small change to $\mathring{\gamma}_p$, and hence a small change to the value of the IGF.

However, there are two cases where discontinuity can occur. The first case of discontinuity is referred to as *path discontinuity*. In this case, with a slight change of a variable base point p from p_1 to p_2, the shortest p-based geodesic loop jumps to a different configuration, as shown in Figure 12. That is, the shortest p_1-based geodesic loop and the shortest p_2-based geodesic loop are far apart from each other. In this case, however, the *length* of $\mathring{\gamma}_p$ still changes continuously; hence, the IGF itself does not experience any discontinuity. Therefore, the path discontinuity does not necessarily mean the discontinuity of the IGF.

In the second case, typically around the extremities of an object, a slight change of a variable base point p causes both a jump of the shortest p-based geodesic loop $\mathring{\gamma}_p$ and discontinuity of the IGF, as shown in Figure 12, where p changes from p_3 to p_4. More specifically, at one point of the process of p moving from p_3 to p_4, $\mathring{\gamma}_p$ will suddenly contract to a point (using a rubber band analogy). Then we will have to find a much longer loop to define the p-based geodesic loop, such as the one based at p_4. This type of discontinuity is referred to as *length discontinuity*, which means the discontinuity of the IGF. Since length discontinuity always occurs, and only occurs, at the extremities of an 3D object, we propose to explore it for tip detection.

Now we consider how to detect the discontinuity of the IGF on a mesh surface in order to effectively identify shape extremities. When a base point p moves from the mesh vertex v_1 to the next vertex v_2, if the p-based geodesic loop changes smoothly, then we have $|I(v_1) - I(v_2)| \leq 2\|\overline{v_1 v_2}\|$, since the composite loop $\mathring{\gamma}_{v_1} \cup \overline{v_1 v_2} \cup \overline{v_2 v_1}$ gives an upper bound on the length of $\mathring{\gamma}_{v_2}$. However, if length discontinuity occurs, we will have $|I(v_1) - I(v_2)| > 2\|\overline{v_1 v_2}\|$, because of the sudden jump of the shortest v_1-based geodesic loop to the much longer v_2-based geodesic loop. Hence, it is suitable to apply the criterion that an edge $\overline{v_1 v_2}$ intersects the *discontinuity contour* of the IGF if

$$|I(v_1) - I(v_2)| > 2\|\overline{v_1 v_2}\|. \tag{2}$$

We apply this criterion to all the edges of an input object (represented as a mesh surface) to find those edges that cross discontinuity contours, and simply use the edge sequences to define the discontinuity contours to delineate the tip parts of the input object.

Figure 13 shows an example of applying the IGF to tip detection, in comparison with the SDF [Gal et al. 2007], the GeoAverage [Martinek et al. 2012], and the persistent heat maxima [Dey et al. 2010], with close-up views of the tip parts to be detected. The previous shape descriptors are based on a continuous function that gradually changes everywhere on the surface of a smooth 3D shape, though they may have a large gradient around a tip region. Therefore, a user-specified parameter needs to be provided in order for these methods to delineate a tip region after the center point of the region is found [Dey et al. 2010]. In contrast, the IGF works with no need for user-specified parameters. However, the discontinuity contour induced by the IGF is sensitive to geometric variations,

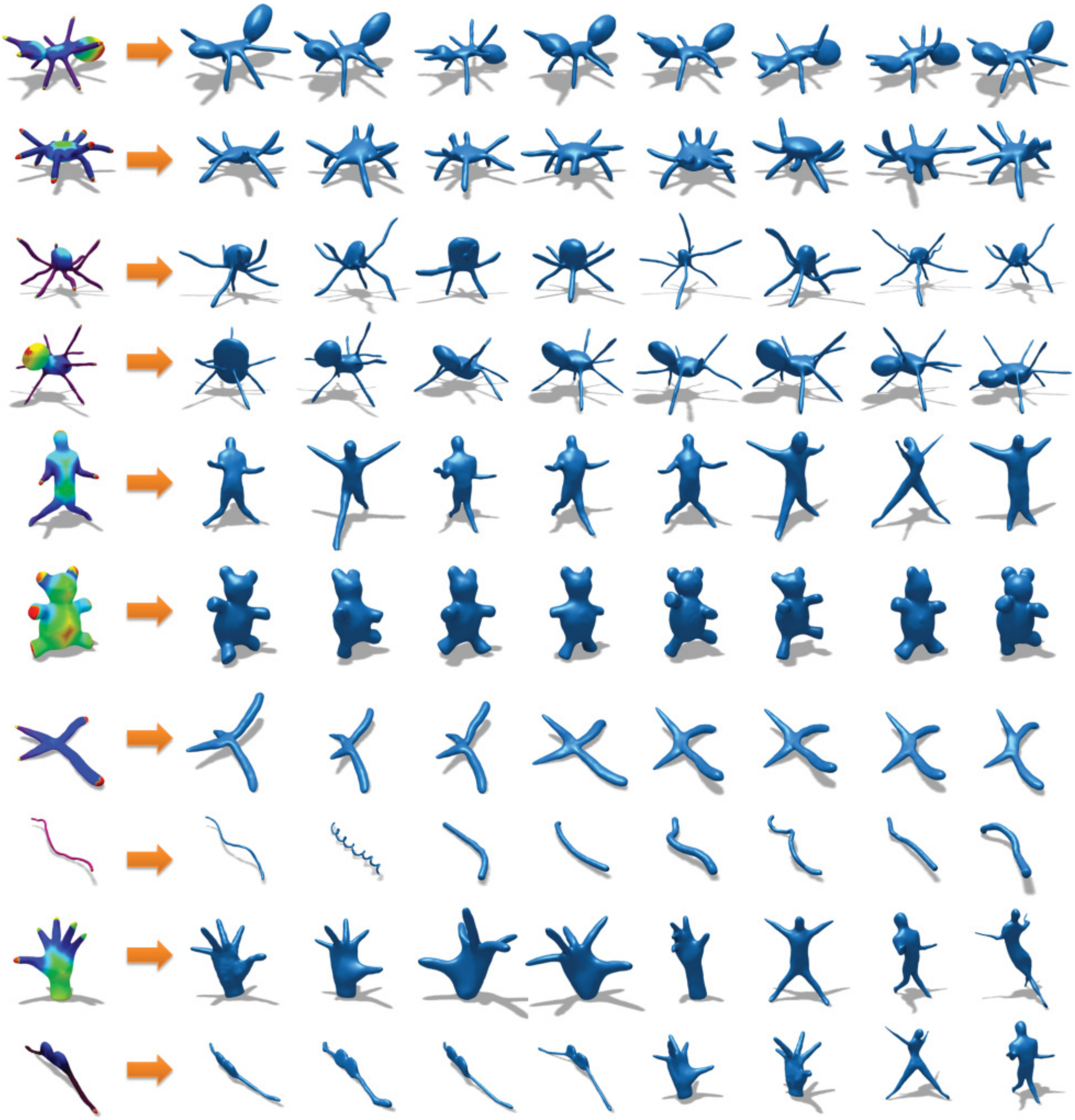


Fig. 11. Shape retrieval examples using the IGF on the articulated models from the McGill 3D Shape Benchmark.

that is, a small perturbation may cause a significant change of the discontinuity contour; see the highlighted boxes in Figure 13.

6.3 Tube/Plate Structure Identification

We now discuss another application of the IGF to shape feature analysis: *identification of tubular parts*. Tubular structures are abundant: bars, tree branches, blood vessels, and so on. Automatic and reliable detection of tubular structures is useful to many geometric processing tasks. For example, in model preparation for 3D printing, the tubular parts of an object, such as thin slivers and seams, need to be identified and checked for strain analysis, since they often represent weak links of the object [Luo et al. 2012; Zhou et al. 2013; Lu et al. 2014]. In biomedical engineering, rodlike structures in trabecular bone need to be identified in establishing correlation between bone architecture and its mechanical characteristics.

Several methods have been proposed for tube or plate structure identification. For example, Goswami et al. [2006] propose to distinguish flat regions from tubular regions of a 3D object by detecting unstable manifolds of index 1 and index 2 saddle points lying on

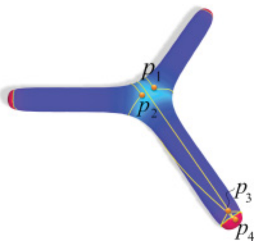


Fig. 12. Path discontinuity and length discontinuity. Path discontinuity means that the shortest p-based geodesic loop jumps to a different configuration but keeps the continuity of IGF when the base point p is enforced a slight change, for example., moving from p_1 to p_2. Length discontinuity means the discontinuity of the IGF, which happens around tip parts, for example,, when p moves from p_3 to p_4.

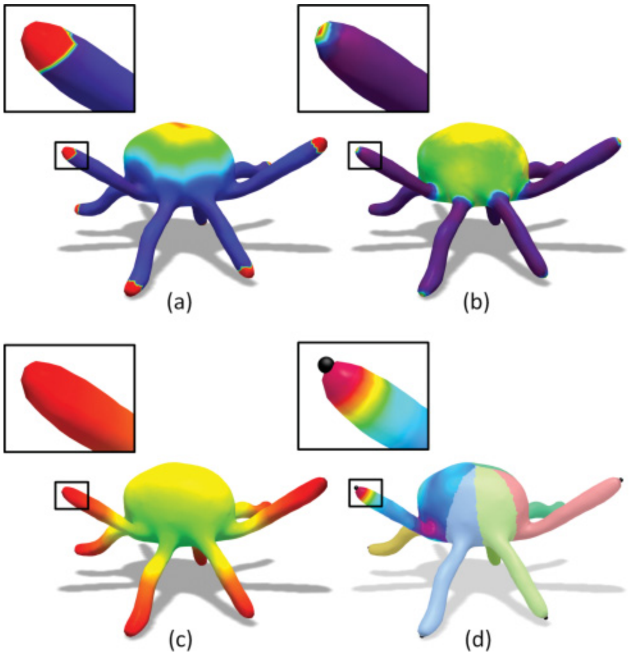


Fig. 13. The IGF has a contour of discontinuity around the tip parts (a), in contrast to the SDF (b), the GeoAverage (c) and the persistent heat maxima (d), which are smooth everywhere.

the interior medial axis of the object. However, their algorithm may misidentify a tubular structure as flat or a flat structure as tubular due to lack of sampling. The method in Wong and Chung [2008]; Bas and Erdogmus [2011] tackles the same problem by extracting model centerlines, followed by principal component analysis (PCA). However, the method requires an expensive voxelization step and cannot resolve intersections between two tubular shapes.

The IGF provides a suitable geometric characterization for identifying tubular structures, because for a point located on a tube its shortest p-based geodesic loop $\mathring{\gamma}_p$ always goes around the tube and gives a relatively small IGF value compared to the length of the tube; and $\mathring{\gamma}_p$ tends to be considerably longer if p is not on a tube. Considering that the SDF [Gal et al. 2007] measures the thickness of an object, we can distinguish plate structures from tubular shapes by combining the SDF and the IGF, that is, defining the plate part as the point set that satisfies

$$\{p \mid \mathrm{IGF}(p) > \tau_1, \mathrm{SDF}(p) < \tau_2\}$$

and the tube part as

$$\{p \mid \mathrm{IGF}(p) < \tau_1, \mathrm{SDF}(p) < \tau_2\},$$

where τ_1, τ_2 are user-specified thickness parameters.

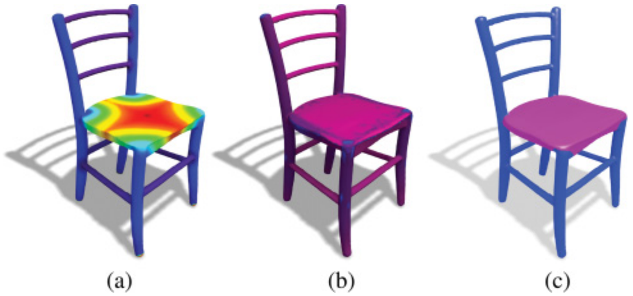


Fig. 14. Combining the IGF (a) and the SDF (b) of the same model, we can distinguish the chair seat (c) from tubular shapes based on user-specified thickness parameters.

Figures 14(a) and (b) respectively show the IGF and the SDF of the Chair model. Note that all the bars and legs exhibit small IGF values and small SDF values, while the chair seat as a plate exhibits large IGF values and small SDF values. Therefore, we can distinguish the seat from tubular shapes by setting proper thickness parameters (see Figure 14(c)). More examples are available in Figure 15.

7. CONCLUSIONS AND DISCUSSION

We have proposed the concepts, algorithms, and applications of the IGF. The IGF is in general defined by the shortest p-based geodesic loop at a point and is therefore invariant under isometry and thus pose oblivious. Tests on the SHREC'11 dataset show that the IGF, when combined with SDF, defines a desirable shape signature. Also, we have shown that the IGF can be applied to detecting tip parts and tube/plate structures of 3D shapes.

Further efforts are needed to improve the efficiency of computing the IGF. We observe that the input surface can be decomposed into separate regions based on the IGF continuity, and inside each region the shortest geodesic loop based at one point can slide continuously to that based at another point, using the rubber-band shrinking technique [Xin et al. 2012; Wu and Tai 2010]. This observation suggests a potential speed-up solution—first computing the IGF on a simplified model and then transferring the IGF from the coarse mesh to the dense version according to the vertex coherence, followed by refinement using the rubber-band shrinking technique. We will investigate this speed-up technique in our future work.

Our algorithm for computing the IGF, in its current form, cannot handle meshes with holes and gaps. Besides, our algorithm works effectively only with a limited range of noise, that is, when vertex perturbation is less than one half of the average mesh edge length. These limitations should be addressed in future works because in practice one often encounters 3D model representations that have topological errors or large geometric noise.

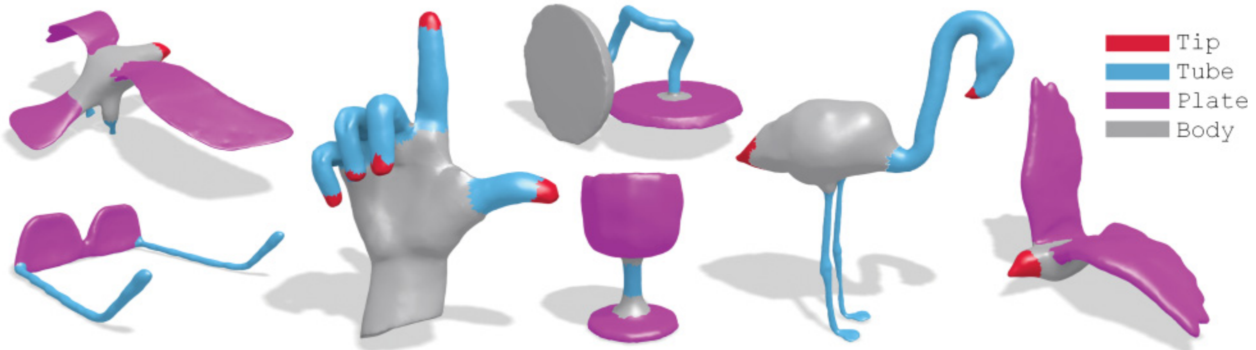


Fig. 15. More examples on detection of tips, tubes and plates.

APPENDIX

A. PROOFS OF LEMMA 4.1 AND LEMMA 4.2

LEMMA 4.1 *Suppose the window w popped out from the priority queue is a pseudosource window rooted at v. Let v' be a one-ring neighbor vertex of v. The situation that the loop $\mathring{\gamma}_p = \gamma(p, v) \cup \overline{vv'} \cup \gamma(v', p)$ defines $I(p)$ belongs to Case 1 if the following conditions are met:*

(1) $\ell(p, v') \leq \ell(p, v)$;
(2) $\mathring{\gamma}_p$ *is locally shortest at v and v';*
(3) $\|\mathring{\gamma}_p\|$ *is shorter than the best-so-far p-based geodesic loop.*

PROOF. When the pseudosource window w rooted at v is taken out from the priority queue, the wavefront just arrives at v, and hence $R_g = \ell(p, v)$. Let $q \in \overline{vv'}$ be the point with the same distance to p in two opposite directions. According to Case 1, we can consider the possibility that $\mathring{\gamma}_p = \gamma(p, v) \cup \overline{vv'} \cup \gamma(v', p)$ defines $I(p)$ when the wavefront arrives at v or v'. If $\ell(p, v') > \ell(p, v)$, then it is not late to consider such a loop until $R_g = \ell(p, v')$, that is, when the pseudosource window rooted at v' is popped out from the priority queue. Therefore, it is reasonable to require $\ell(p, v') \leq \ell(p, v)$. The second and last conditions are obvious. ☐

LEMMA 4.2 *Suppose the window w popped out from the priority queue is an interval window arriving at the edge $e = \overline{v_1 v_2}$, and w^- is an interval window on the reverse edge e^-. The situation that the windows w and w^- define $I(p)$ belongs to Case 2 or Case 3 if*

(1) $\ell_{max}(w^-) + 2\max(\|v_1 v\|, \|v_2 v\|) \geq \ell_{min}(w)$, *where v is the vertex opposite to e, and ℓ_{max} and ℓ_{min} respectively denote the maximum and minimum geodesic distances given by an interval window;*
(2) The straight line by connecting w and w^-'s roots goes through w and w^-'s common interval;
(3) The new loop $\mathring{\gamma}_p$ is shorter than the best-so-far p-based geodesic loop.

PROOF. It is easy to See that $R_g = \ell_{min}(w)$ when w is popped out from the priority queue. Let q be the point at the same distance to p in two opposite directions. Case 2 or Case 3 (see Figure 5) implies that q must be inside the face incident to e or e^-. With loss of generality, we assume that q lies in the face incident to e, that is, the triangle face $\triangle v v_1 v_2$. Since it is sufficient to consider the possibility of forming the shortest p-based geodesic loop when either w or w^- is popped out from the priority queue, we consider w in this case. Obviously, we have

$$\ell_{\min}(w) \leq \ell(p, q) + \max(\|v_1 v\|, \|v_2 v\|)$$

and

$$\ell_{\max}(w^-) \geq \ell(p, q) - \max(\|v_1 v\|, \|v_2 v\|).$$

Combining them together, we prove the necessity of the first condition. The second and last conditions are straightforward. ☐

ACKNOWLEDGMENTS

We are grateful to Professor T. K. Dey for his kind help with our work on tip detection. The models in this article are provided courtesy of the AIM@SHAPE Shape Repository, the SHREC'11 dataset and the McGill 3D Shape Benchmark.

REFERENCES

Uwe Abresch and Wolfgang T. Meyer. 1997. Injectivity radius estimates and sphere theorems. *Karsten Grove* 30 (1997), 1–47.

J. Assfalg, M. Bertini, A. D. Bimbo, and P. Pala. 2007. Content-based retrieval of 3D objects using spin image signatures. *IEEE Trans. Multimed.* 9, 3 (2007), 589–599.

Mathieu Aubry, Ulrich Schlickewei, and Daniel Cremers. 2011. The wave kernel signature: A quantum mechanical approach to shape analysis. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 1626–1633.

Erhan Bas and Deniz Erdogmus. 2011. Principal curves as skeletons of tubular objects. *Neuroinformatics* 9, 2–3 (2011), 181–191.

Serge Belongie, Jitendra Malik, and Jan Puzicha. 2000. Shape context: A new descriptor for shape matching and object recognition. In *NIPS*. 831–837.

Alexander M. Bronstein, Michael M. Bronstein, Leonidas J. Guibas, and Maks Ovsjanikov. 2011. Shape Google: Geometric words and expressions for invariant shape retrieval. *ACM Trans. Graph.* 30, 1 (2011), 1:1–1:20.

Michael M. Bronstein and Iasonas Kokkinos. 2010. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, Washington DC, 1704–1711.

Ee Chien Chang, Zhiyong Huang, Mohan S. Kankanhalli, and Rong Xu. 2003. A 3D shape matching framework. In *Proceedings of the 1st International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia (GRAPHITE'03)*. ACM, New York NY, 269–270.

Jindong Chen and Yijie Han. 1990. Shortest paths on a polyhedron. In *Proceedings of the 6th Annual Symposium on Computational Geometry*. ACM, New York, NY, 360–369.

Yee Ming Chen and Jen-Hong Chiang. 2010. Fusing multiple features for Fourier Mellin-based face recognition with single example image per person. *Neurocomputing* 73, 16 (2010), 3089–3096.

Hyeong In Choi, Sung Woo Choi, and Hwan Pyo Moon. 1997. Mathematical theory of medial axis transform. *Pac. J. Math.* 181, 1 (1997), 57–88.

Keenan Crane, Clarisse Weischedel, and Max Wardetzky. 2013. Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Trans. Graph.* 32, 5 (2013), 152.

Tamal K. Dey, Kuiyu Li, Chuanjiang Luo, Pawas Ranjan, Issam Safa, and Yusu Wang. 2010. Persistent heat signature for pose-oblivious matching of incomplete models. *Comput. Graph. Forum* 29, 5 (2010), 1545–1554.

Lubin Fan, Ligang Liu, and Kun Liu. 2011. Paint mesh cutting. *Comput. Graph. Forum* 30, 2 (2011), 603–611.

R. Gal, A. Shamir, and D. Cohen-Or. 2007. Pose-oblivious shape signature. *IEEE Trans. Visualiz. Comput. Graph.* 13, 2 (2007), 261–271.

Timothy Gatzke, Cindy Grimm, Michael Garland, and Steve Zelinka. 2005. Curvature maps for local shape comparison. In *Proceedings of the International Conference on Shape Modeling and Applications*. IEEE, 244–253.

Samrat Goswami, Tamal K. Dey, and Chandrajit L. Bajaj. 2006. Identifying flat and tubular regions of a shape by unstable manifolds. In *Proceedings of the 2006 ACM Symposium on Solid and Physical Modeling*. ACM, New York, NY, 27–37.

Eberhard Hopf. 1948. Closed surfaces without conjugate points. *Proc. Natl. Acad. Sci. U.S.A.* 34, 2 (1948), 47.

Jin Huang, Muyang Zhang, Jin Ma, Xinguo Liu, Leif Kobbelt, and Hujun Bao. 2008. Spectral quadrangulation with orientation and alignment control. *ACM Trans. Graph.* 27, 5, Article 147, 9 pages. `DOI:`http://dx.doi.org/10.1145/1409060.1409100

Adrian Ion, Nicole M. Artner, Gabriel Peyré, Salvador B López Mármol, Walter G. Kropatsch, and Laurent Cohen. 2008. 3D shape matching by geodesic eccentricity. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 1–8.

Adrian Ion, Gabriel Peyré, Yll Haxhimusa, Samuel Peltier, Walter G. Kropatsch, Laurent D. Cohen, et al. 2007. Shape matching using the geodesic eccentricity transform-a study. In *Proceedings of the 31st Annual Workshop of the Austrian Association for Pattern (OAGM/AAPR)*. 97–104.

I. M. James. 1962. Convex regions in the geometry of paths. In *Differential Geometry*. Pergamon, London, 223–232.

Kalervo Järvelin and Jaana Kekäläinen. 2000. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, New York, NY, 41–48.

Andrew E. Johnson and Martial Hebert. 1999. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Analy. Mach. Intell.* 21, 5 (1999), 433–449.

Ron Kimmel and James A. Sethian. 1998. Computing geodesic paths on manifolds. *Proc. Natl. Acad. Sci. U.S.A* 95, 15 (1998), 8431–8435.

Zhenzhong Kuang, Zongmin Li, Xiaxia Jiang, Yujie Liu, and Hua Li. 2015. Retrieval of non-rigid 3D shapes from multiple aspects. *Computer-Aided Des.* 58 (2015), 13–23.

Zhouhui Lian, Afzal Godil, Benjamin Bustos, Mohamed Daoudi, Jeroen Hermans, Shun Kawamura, et al. 2011. SHREC'11 track: Shape retrieval on non-rigid 3D watertight meshes. *3DOR* 11 (2011), 79–88.

Zhouhui Lian, Afzal Godil, Benjamin Bustos, Mohamed Daoudi, Jeroen Hermans, Shun Kawamura, et al. 2013. A comparison of methods for non-rigid 3D shape retrieval. *Pattern Recogn.* 46, 1 (2013), 449–461.

Roee Litman, Alex Bronstein, Michael Bronstein, and Umberto Castellani. 2014. Supervised learning of bag-of-features shape descriptors using sparse coding. *Comput. Graph. Forum* 33, 5, 127–136.

Yong-Jin Liu. 2013. Exact geodesic metric in 2-manifold triangle meshes using edge-based data structures. *Computer-Aided Des.* 45, 3 (2013), 695–704.

Lin Lu, Andrei Sharf, Haisen Zhao, Yuan Wei, Qingnan Fan, Xuelin Chen, Yann Savoye, Changhe Tu, Daniel Cohen-Or, and Baoquan Chen. 2014. Build-to-last: Strength to weight 3D printed objects. *ACM Trans. Graph.* 33, 4 (2014), 1–10.

Linjie Luo, Ilya Baran, Szymon Rusinkiewicz, and Wojciech Matusik. 2012. Chopper: Partitioning models into 3D-printable parts. *ACM Trans. Graph.* 31, 6 (2012), 129.

Michael Martinek, Matthias Ferstl, and Roberto Grosso. 2012. 3D shape matching based on geodesic distance distributions. In *Vision, Modeling & Visualization*. The Eurographics Association, 219–220.

Ivan Mendoza. 2011. Local features for partial shape matching and retrieval. In *Proceedings of the 19th ACM International Conference on Multimedia (MM'11)*. ACM, New York, NY, 853–856.

J. S. B. Mitchell, D. M. Mount, and C. H. Papadimitriou. 1987. The discrete geodesic problem. *SIAM J. Comput.* 16, 4 (1987), 647–668.

Robert Osada, Thomas Funkhouser, Bernard Chazelle, and David Dobkin. 2002. Shape distributions. *ACM Trans. Graph.* 21, 4 (2002), 807–832.

Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. 2012. Functional maps: A flexible representation of maps between shapes. *ACM Trans. Graph.* 31, 4 (2012).

Ofir Pele and Michael Werman. 2009. Fast and robust earth mover's distances. In *IEEE 12th International Conference on Computer Vision*. IEEE, Washington, DC, 460–467.

Konrad Polthier and Markus Schmies. 1999. Geodesic flow on polyhedral surfaces. In *Data Visualization'99*. Springer, Berlin, 179–188.

Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. 2000. The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vision* 40, 2 (2000), 99–121.

Ariel Shamir, Lior Shapira, Daniel Cohen-Or, and Rony Goldenthal. 2004. Geodesic mean shift. In *Proceedings of 5th Korea Israel Conference on Geometric Modeling and Computer Graphics*. 51–56.

Lior Shapira, Shy Shalom, Ariel Shamir, Daniel Cohen-Or, and Hao Zhang. 2010. Contextual part analogies in 3D objects. *Int. J. Comput. Vision* 89, 2–3 (2010), 309–326.

Lior Shapira, Ariel Shamir, and Daniel Cohen-Or. 2008. Consistent mesh partitioning and skeletonisation using the shape diameter function. *Vis. Comput.* 24, 4 (2008), 249–259.

M. Sharir and A. Schorr. 1986. On shortest paths in polyhedral spaces. *SIAM J. Comput.* 15, 1 (1986), 193–215.

Philip Shilane, Patrick Min, Michael Kazhdan, and Thomas Funkhouser. 2004. The Princeton shape benchmark. In *Proceedings of the Shape Modeling International (SMI'04)*. 167–178.

Kaleem Siddiqi, Juan Zhang, Diego Macrini, Ali Shokoufandeh, Sylvain Bouix, and Sven Dickinson. 2008. Retrieving articulated 3D models using medial surfaces. *Mach. Vis. Appl.* 19, 4 (2008), 261–275.

Justin Solomon, Raif Rustamov, Leonidas Guibas, and Adrian Butscher. 2014. Earth mover's distances on discrete surfaces. *ACM Trans. Graph.* 33, 4 (July 2014), 67:1–67:12.

Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. 2009. A concise and provably informative multi-scale signature based on heat diffusion. *Comput. Graph. Forum* 28, 5 (2009), 1383–1392.

Vitaly Surazhsky, Tatiana Surazhsky, Danil Kirsanov, Steven J. Gortler, and Hugues Hoppe. 2005. Fast exact and approximate geodesics on meshes. *ACM Trans. Graph.* 24, 3 (2005), 553–560.

Johan W. H. Tangelder and Remco C. Veltkamp. 2008. A survey of content based 3D shape retrieval methods. *Multimed. Tools Appl.* 39, 3 (2008), 441–471.

Wilbur C. K. Wong and Albert Chung. 2008. Principal curves to extract vessels in 3D angiograms. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'08)*. 1–8.

Chunlin Wu and Xuecheng Tai. 2010. A level set formulation of geodesic curvature flow on simplicial surfaces. *IEEE Trans. Visualiz. Comput. Graph.* 16, 4 (2010), 647–662.

Ying He, Xiang Ying, and Shi-Qing Xin. 2014. Parallel Chen-Han (PCH) algorithm for discrete geodesics. *ACM Trans. Graph.* 33, 1 (2014), 57–76.

Shi-Qing Xin, Ying He, and Chi-Wing Fu. 2012. Efficiently computing exact geodesic loops within finite steps. *IEEE Trans. Visualiz. Comput. Graph.* 18, 6 (2012), 879–889.

Shi-Qing Xin and Guo-Jin Wang. 2009. Improving Chen and Han's algorithm on the discrete geodesic problem. *ACM Trans. Graph.* 28, 4, Article 104, 8 pages.

Xiang Ying, Xiaoning Wang, and Ying He. 2013. Saddle vertex graph (SVG): A novel solution to the discrete geodesic problem. *ACM Trans. Graph.* 32, 6 (2013), 170.

Qingnan Zhou, Julian Panetta, and Denis Zorin. 2013. Worst-case structural analysis. *ACM Trans. Graph.* 32, 4, Article 137, 12 pages.

Henrik Zimmer, Marcel Campen, and Leif Kobbelt. 2013. Efficient computation of shortest path-concavity for 3D meshes. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'13)*. 2155–2162.

Received October 2014; revised August 2015; accepted December 2015