Shape Segmentation and Matching with Flow Discretization *

Tamal K. Dey
Dept. of CIS
The Ohio State U.
Columbus, Ohio 43081
USA

Joachim Giesen
Theoretische Informatik
ETH Zürich
CH-8092 Zürich
Switzerland

Samrat Goswami
Dept. of CIS
The Ohio State U.
Columbus, Ohio 43081
USA

Abstract

Geometric shapes are identified with their features. For computational purposes a concrete mathematical definition of features is required. In this paper we use a topological approach, namely dynamical systems, to define features of shapes. To exploit this definition algorithmically we assume that a point sample of the shape is given as input from which features of the shape have to be approximated. We translate our definition of features to the discrete domain while mimicking the set-up developed for the continuous shapes. The outcome of this approach is a clean mathematical definition of features that are efficiently computable with combinatorial algorithms. Experimental results show that our algorithms segment shapes in two and three dimensions into so-called features quite effectively. Further, we develop a shape matching algorithm that takes advantage of our robust feature segmentation step. Performance of this algorithm is exhibited with experimental results.

1. Introduction

The features of a shape are its specific identifiable subsets. Although this high level characterization of features is assumed routinely, more concrete and mathematical definitions are required for computational purposes. Many applications including object recognition, classification, matching, tracking need to solve the problem of segmenting a shape into its salient features, see for example [9, 10, 17, 23, 28, 35]. Most of these applications need an appropriate definition of features that are computable. In the computational domains, often the shapes are represented with discrete means that approximate them. Consequently, a consistent definition of features in the discrete domain is needed to compute them reliably.

Different geometric and topological structures such as shock graphs [31, 29], medial axes [22] and Reeb graphs [17] have been proposed in the past for shape segmentation. Two notable topological approaches related to shape features are level sets method [30] and the topological persistence [13]. The level sets method use numerical techniques to compute features whereas we rely more on combinatorial means. This makes computations faster and more robust against numerical errors. Topological persistence method works with homological algebra to compute a signature of the shape that respects its features, but do not address the segmentation issue.

In this paper we use a topological approach, namely dynamical systems, to define features of shapes. This approach has been studied in the context of surface reconstruction recently [12, 15]. We assume that a point sample of the shapes is given as input from which features of the shape have to be approximated. We translate our definition of features to this discrete domain while mimicking the set-up that we develop in the continuous case. The outcome of this approach is a clean mathematical definition of features that are computable with combinatorial algorithms. For shapes in the plane we compute them exactly whereas we approximate them for shapes embedded in $\mathbb{R}^3$ mimicking the two dimensional algorithm. Our experimental results show that our algorithms segment shapes in two and three dimensions into so-called features quite effectively.

We apply our feature segmentation technique to the shape matching problem, where a similarity measure is sought between two shapes. An usual approach in shape matching is to compute a signature of a shape and then comparing it with the signature of the other shape. Different quantities such as curvature distribution [3, 32], wavelet coefficients [19], Fourier descriptors [2], geometric statistics [4, 33], spin image [20] and shape distribution [24] have been suggested for shape signatures; see survey articles [1, 7, 23, 34] for more details. Another prevalent approach is to segment a shape into its salient features and then match the shapes based on the features and their spatial relationships [5, 6, 8, 16]. These feature based approaches depend mainly on the quality of the feature detection step.

*This work is partially supported by NSF under grant DMS-0138456 with a subcontract from Stanford University).

We give a shape matching algorithm that takes the advantage of our robust feature segmentation step. Each significant feature segment is represented by a weighted point where the weight is the volume of the segment. Then, the shape matching problem boils down to matching two small weighted point sets instead of matching large point sets derived from the boundary of the shapes [18]. We carry out these steps so that the entire matching process remains invariant to rotation, translation, mirroring and scaling.

2. Flow and critical points

In shape segmentation and shape matching we deal with continuous shapes Σ. Typically these shapes are bounded by one or two dimensional manifolds embedded in $\mathbb{R}^2$ or $\mathbb{R}^3$ respectively. In this section we outline a theory of the flow induced by a shape. Later we will use this theory to define and compute features of shapes. Here we will develop the theory in a more general setting by considering general shapes embedded in d-dimensional Euclidean space $\mathbb{R}^d$.

Height function. In the following Σ always denotes a compact subset of $\mathbb{R}^d$. The set Σ can be used to define a height function $h : \mathbb{R}^d \to \mathbb{R}$ as follows:

$$h(x) \; = \; \inf_{p\in\Sigma} \|p - x\|^2 \;\text{ for all } x \in \mathbb{R}^d.$$

Anchor set. Associated with the height function, we define an *anchor set* for each point $x \in \mathbb{R}^d$ as follows:

$$A(x) \; = \; \operatorname{argmin}_{p\in\Sigma} \|p - x\|$$

Basically, $A(x)$ is the set of closest point to x in Σ; see Figure 1. Note that $A(x)$ can contain even a continuum of points.

We would like to define a unit vector field $v : \mathbb{R}^d \to \mathbb{R}^d$ that assigns to every point $x \in \mathbb{R}^d$ the direction in which the height function increases the most. If h is smooth at x then $v(x)$ coincides with the normalized gradient $\nabla h(x)/\|\nabla h(x)\|$. In our case h is not smooth everywhere. So, we have to be careful to define $v(x)$ at any non-smooth point x. Instead of smooth and non-smooth points we will talk about regular and critical points in the following. Critical points are either local extrema or saddle points of the height function. We use a generalized theory of critical points, see for example [25], to derive the following definition.

Regular and critical point. For every point $x \in \mathbb{R}^d$ let $H(x)$ be the convex hull of $A(x)$, i.e. the convex hull of the points on Σ that are closest to x. We call x a critical point of h if $x \in H(x)$. Otherwise we call x a regular point.

The following definition turns out to be very helpful in the subsequent discussion. It allows us to characterize the direction of steepest ascent of the height function h at every point $x \in \mathbb{R}^d$.

Driver. For any point $x \in \mathbb{R}^d$ let $d(x)$ be the point in $H(x)$ closest to x. We call $d(x)$ the driver of x.

Lemma 1 *For any regular point $x \in \mathbb{R}^d$ let $d(x)$ be the driver of x. The steepest ascent of the height function h at x is in the direction of $x - d(x)$.* $\square$

PROOF. Though h is not smooth at x it is still continuous. Thus in the limit the direction of steepest ascent of h at x is determined by the distance of x from $A(x)$.

Our assumption that x is a regular point, i.e. $x \notin H(x)$, implies that $d(x)$ is contained in the boundary of $H(x)$. That is, the vector $x - d(x)$ is non-zero. We want to show that h is non-increasing in all directions that make an angle of $\pi/2$ or larger with $x - d(x)$.

By definition all points in $A(x)$ have the same distance d from x and $|A(x)| \geq 1$. For any $p \in A(x)$ let H_p be the unique closed halfspace that has x in its boundary and contains the ball of radius d centered at p. The distance function h at x is non-increasing in all directions that point into H_p for all $p \in A(x)$. Let

$$H \; = \; \bigcup_{p\in A(x)} H_p,$$

By construction all directions at x that make an angle of $\pi/2$ or larger with $x - d(x)$ point into H. Thus h is non-increasing in all that directions. This implies that h can only be increasing in some direction v if $\langle v, x - d(x)\rangle > 0$. Since one can decompose any direction into its part along $x - r$ and its part orthogonal to it this shows that h has its steepest ascent at x in direction $x - d(x)$. $\square$

We are now going to use the direction of steepest ascent to define a flow on $\mathbb{R}^d$, i.e. a dynamical system on $\mathbb{R}^d$.

Induced flow. Define a vector field $v : \mathbb{R}^d \to \mathbb{R}^d$ by setting

$$v(x) \; = \; \frac{x - d(x)}{\|x - d(x)\|} \;\text{ if } x \neq d(x) \text{ and } 0 \text{ otherwise.}$$

The flow induced by the vector field v is a function $\phi : [0,\infty) \times \mathbb{R}^d \to \mathbb{R}^d$ such that the right derivative at every point $x \in \mathbb{R}^d$ satisfies the following equation

$$\lim_{t\downarrow t_0} \frac{\phi(t,x) - \phi(t_0,x)}{t - t_0} \; = \; v(\phi(t_0,x)).$$

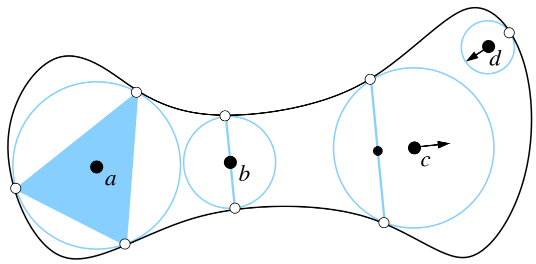


Figure 1: In this example Σ is a curve embedded in $\mathbb{R}^2$. Shown are four points $x = a, b, c, d \in \mathbb{R}^2$, the corresponding sets $A(x)$ are hollow circles, the convex hull of $A(x)$ are light shaded and the driver of the point c is the smaller black circle. The driver of the point d is the single point in $A(d)$. The points a and b are critical since they are contained in $H(a)$ and $H(b)$, respectively. The points c and d are regular. The direction of steepest ascent of the height function at c and d is indicated by an arrow.

Orbits and fixpoints. Given $x \in \mathbb{R}^d$ and an induced flow ϕ, the curve $\phi_x : [0, \infty) \to \mathbb{R}^d$, $t \mapsto \phi(t, x)$ is called the *orbit* of x. A point $x \in \mathbb{R}^d$ is called a *fixpoint* of ϕ if $\phi(t, x) = x$ for all $t \geq 0$.

Basically, the orbit of a point is the curve it will follow if it were let move along the flow.

Observation 1 *The fixpoints of ϕ are the critical points of the height function h.*

Because of this observation we refer to a fixpoint of ϕ as a minimum, saddle or maximum if the corresponding critical point of the height function is a minimum, saddle or maximum, respectively.

Stable manifold. The stable manifold $S(x)$ of a critical point x is the set of all points that flow into x, i.e.

$$S(x) = \{y \in \mathbb{R}^d : \lim_{t\to\infty} \phi_y(t) = x\}.$$

The stable manifolds of all critical points partition $\mathbb{R}^d$, i.e.

$$\mathbb{R}^d = \bigcup_{\text{critical points } x} S(x)$$

and $S(x) \cap S(y) = \emptyset$ for any two different critical points x and y.

3. Discretization

To deal with continuous shapes algorithmically we discretize them. Here discretization means taking a finite sample P of the shape $\Sigma \subset \mathbb{R}^d$. That is, we replace Σ by a finite subset of Σ. The sample P induces another vector field which resembles the vector field induced by Σ provided P is sufficiently dense in Σ. The vector field induced by P is intimately linked with the Voronoi- and the Delaunay diagram of P. Moreover, the stable manifolds corresponding to the flow induced by this vector field are efficiently computable in dimensions two and three.

Let us first summarize the definitions of Voronoi- and Delaunay diagrams before we show how the concepts we introduced in the last section can be specialized to the case of finite point sets.

Voronoi diagram. Let P be a finite set of points in $\mathbb{R}^d$. The *Voronoi cell* of $p \in P$ is given as

$$V_p = \{x \in \mathbb{R}^d : \forall q \in P - \{p\}, \|x - p\| \leq \|x - q\|\}.$$

The sets V_p are convex polyhedra or empty since the set of points that have the same distance from two points in P forms a hyperplane. Closed facets shared by $k, 2 \leq k \leq d$, Voronoi cells are called $(d - k + 1)$-dimensional *Voronoi facets* and points shared by $d + 1$ or more Voronoi cells are called *Voronoi vertices*. The term *Voronoi object* denotes either a Voronoi cell, facet, edge or vertex. The *Voronoi diagram* V_P of P is the collection of all Voronoi objects. It defines a cell decomposition of $\mathbb{R}^d$.

Delaunay diagram. The *Delaunay diagram* D_P of a set of points P is dual to the Voronoi diagram of P. The convex hull of $d + 1$ or more points in P defines a *Delaunay cell* if the intersection of the corresponding Voronoi cells is not empty and there exists no superset of points in P with the same property. Analogously, the convex hull of $k \leq d$ points defines a $(k - 1)$-dimensional *Delaunay face* if the intersection of their corresponding Voronoi cells is not empty. Every point in P is called *Delaunay vertex*. The term *Delaunay object* denotes either a Delaunay cell, face, edge or vertex. The Delaunay diagram D_P defines a decomposition of the convex hull of all points in P. This decomposition is a triangulation if the points are in general position.

We always refer to the interior and to the boundary of Voronoi-/Delaunay objects with respect to their dimension, e.g. the interior of a Delaunay edge contains all points in this edge besides the endpoints. The interior of a vertex and its boundary are the vertex itself. Furthermore, we always assume general position unless stated differently.

Now consider the height function h as in the previous section but replacing Σ with its discrete sample P. Define critical points for h as we did in the continuous case.

Lemma 2 *Let P be a finite set of points such that Voronoi and their dual Delaunay objects intersect in their interiors if they intersect at all. Then the critical points of the height*

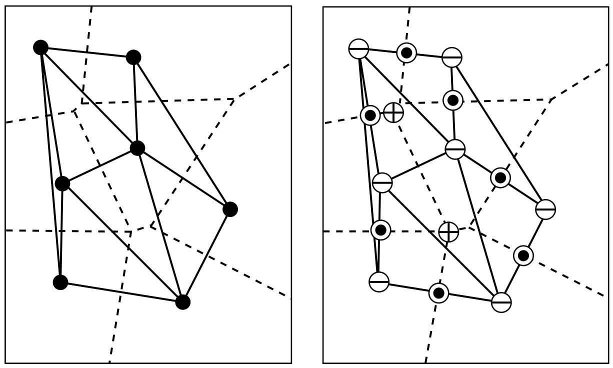


Figure 2: On the left: The Voronoi diagram (dashed lines) and the Delaunay triangulation (solid lines) of seven points in $\mathbb{R}^2$. On the right: The critical points (maxima $\oplus$, saddle points $\odot$ and minima $\ominus$) of the height function induced by the seven points.

function h are the intersection points of Voronoi objects V and their dual Delaunay object σ. □

This characterization of critical points can be used to assign a meaningful index to critical points, namely, the index of a critical point is the dimension of the Delaunay object used in the above characterization. Minima always have index 0 and maxima always have index d.

The driver of a point in $\mathbb{R}^d$ can now also be described in terms of Voronoi- and Delaunay objects.

Lemma 3 *Given $x \in \mathbb{R}^d$. Let V be the lowest dimensional Voronoi object in the Voronoi diagram of P that contains x and let σ be the dual Delaunay object of V. The driver of x is the point on σ closest to x.* □

We have a much more explicit characterization of the flow induced by a finite point set than in the general case.

Observation 2 *The flow ϕ induced by a finite point set P is given as follows: For all critical points x of the height function associated with P we set:*

$$\phi(t,x) = x \ , \ t \in [0,\infty)$$

Otherwise let $d(x)$ be the driver of x and R be the ray originating at x and shooting in the direction $v(x) = x - d(x)/\|x - d(x)\|$. Let z be the first point on R whose driver is different from $d(x)$. Note that such a z need not exist in $\mathbb{R}^d$ if x is contained in an unbounded Voronoi object. In this case let z be the point at infinity in the direction of R. We set:

$$\phi(t,x) = x + t \cdot v(x) \ , \ t \in [0, \|z - x\|)$$

For $t \geq \|z - x\|$ the flow is given as follows:

$$\begin{aligned}\phi(t,x) &= \phi\left(t - \|z-x\| + \|z-x\|, \, x\right) \\ &= \phi\left(t - \|z-x\|, \, \phi\left(\|z-x\|, \, x\right)\right)\end{aligned}$$

It is not completely obvious, but it can be shown that this flow is well defined. It is also easy to see that the orbits of ϕ are piecewise linear curves that are linear in Voronoi objects. See Figure 4 for some examples of orbits.

Under some mild non-degeneracy condition the stable manifolds of the critical points have a nice recursive structure. A stable manifold of index $k, 0 \leq k \leq d$, has dimension k and its boundary is made up from stable manifolds of index $k-1$ critical points.

In $\mathbb{R}^2$ the stable manifolds of index 1 critical points, i.e. saddle points, are exactly the edges of the Gabriel graph of the point set P. The Gabriel graph is efficiently computable. The recursive structure of the stable manifolds now tells us that the stable manifolds of the maxima, i.e. index 2 critical points, are exactly the compact regions of the Gabriel graph. That is, the stable manifolds of maxima (index 2 critical points) are given as a union of Delaunay triangles.

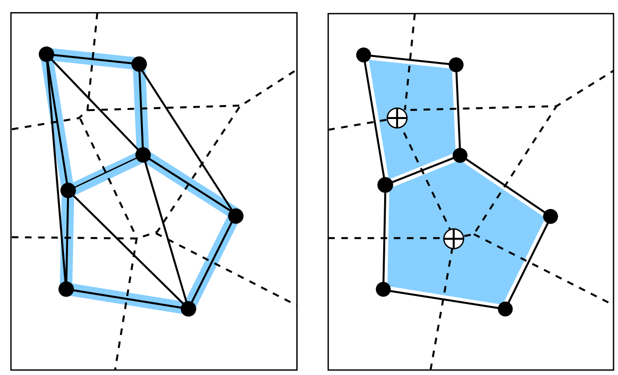


Figure 3: On the left: The edges of the Gabriel graph of the seven points from Figure 1 are highlighted. On the right: The stable manifolds of the maxima $\oplus$ of the flow induced by the seven points.

The stable manifolds of flows induced by finite point sets in $\mathbb{R}^3$ can also be computed efficiently, see [15]. But already in $\mathbb{R}^3$ the stable manifolds of index 2 saddle points and maxima are not given as sub-complexes of the three dimensional Delaunay triangulation. Nevertheless, we will show in the next section that these stable manifolds can be approximated by sub-complexes of the Delaunay triangulation.

4. Approximating stable manifolds

Our goal is to decompose a two or three dimensional shape Σ into disjoint segments that respect the 'features' of the shape. In our first attempt to define features we resort to stable manifolds of maxima. So, we define a feature to be the closed stable manifold $F(x)$ of a maximum x,

$$F(x) = \text{closure}\,(\,S(x)).$$

Figure 5(a) shows the segmentation of a shape in $\mathbb{R}^2$ with this definition of features. We can translate this definition to the discrete setting immediately as we have mimicked all concepts in the continuous case to the discrete setting. Figure 5(b) shows this segmentation.

From a point sample P of a shape Σ we would like to compute $F(x)$ for all maxima x. These maxima are a subset of the Voronoi vertices in V_P. For computing the feature segmentation it is sufficient to compute the boundary of all such $F(x)$. As we observed earlier this boundary is partitioned by the stable manifolds of critical points of lower index. In $\mathbb{R}^2$ this means that Gabriel edges separate the features.

We also want to separate the features in $\mathbb{R}^3$ by a subset of the Delaunay triangles. That is, we want to approximate the boundary of the stable manifolds of maxima by Delaunay triangles. These boundaries are made up from stable manifolds of critical points of index 1 and 2. The closures of the stable manifolds of index 1 critical points are again exactly the Gabriel edges. By Lemma 2 each critical point of index 2 lies in a Delaunay triangle which we call a *saddle* triangle. The stable manifolds of the index 2 critical points may not be contained only in the saddle triangles. This makes computing the boundary of the stable manifolds of maxima harder in $\mathbb{R}^3$. Although it can be computed exactly, we propose an alternative method that approximates this boundary using only Delaunay triangles. We derive this method by generalizing a simple algorithm that computes the closed stable manifolds for maxima in $\mathbb{R}^2$ exactly.

In $\mathbb{R}^2$ we can compute the closed stable manifold $F(x)$ of a maximum x by exploring out from the Delaunay triangle containing x. To explain the algorithm we define a flow relation among Delaunay triangles which was proposed by Edelsbrunner et al. [14] for computing pockets in molecules.

Flow relation in $\mathbb{R}^2$. Let σ_1, σ_2 be two triangles that share an edge e. We say $\sigma_1 < \sigma_2$ if σ_1 and its dual Voronoi vertex lie on the opposite sides of the supporting line of e.

Observation 3 *Let σ_1 and σ_2 be two triangles sharing an edge e where $\sigma_1 < \sigma_2$. Then the flow on the dual Voronoi edge v_1v_2 of e is directed from v_1 to v_2 where v_i is the dual Voronoi vertex of σ_i.*

It is obvious from the definition that the transitive closure $<^*$ of $<$ is acyclic. If $\sigma_1 < \sigma_2$, then the radius of the circumcircle of σ_2 is larger than the radius of the circumcircle of σ_1. So, in a chain of triangles related by $<$ relation the circumradii of the triangles can never decrease, thus making it impossible for $<^*$ to be cyclic. This means that, for each triangle σ', there is a triangle σ containing a maximum x such that $\sigma' <^* \sigma$. We will say that σ' flows into σ.

The following lemma holds in $\mathbb{R}^2$.

Lemma 4 *Let σ be a triangle containing a maximum x. We have $F(x) = \bigcup_{\sigma' <^* \sigma} \sigma'$.* $\square$

The algorithm for computing the closed stable manifold $F(x)$ follows immediately from the above lemma. Initially $F(x)$ is set to the triangle σ that contains x. At any generic step of this exploration, let e be a Delaunay edge that lies on the boundary of $F(x)$ computed so far. Let σ_1 and σ_2 be two triangles that share e where σ_1 is outside $F(x)$. If $\sigma_1 < \sigma_2$ we update $F(x)$ as $F(x) := F(x) \cup \sigma_1$. This process continues till we cannot include any more triangle into $F(x)$.

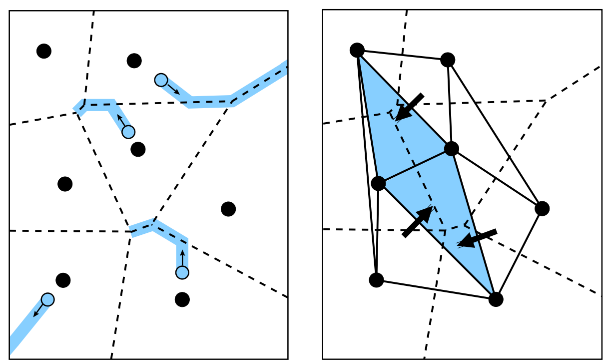
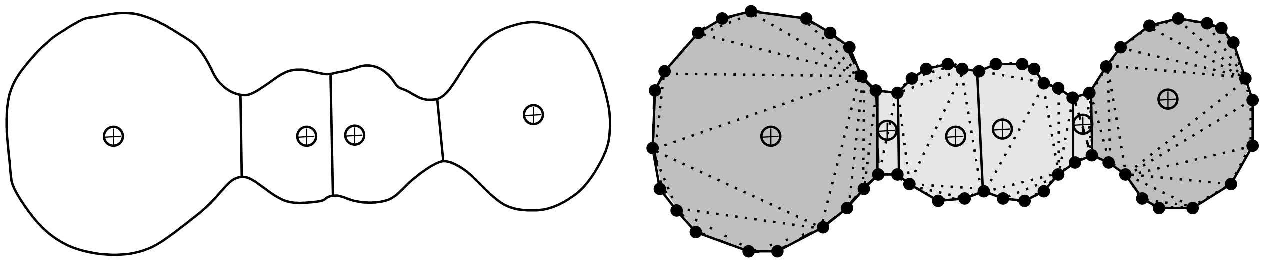


Figure 4: On the left: Some orbits of the flow induced by the points from Figure 2. Note that these orbits are piecewise linear curves. On the right: Triangles containing a maximum are highlighted. An arrow pointing from a triangle σ' to a triangle σ indicates that $\sigma' < \sigma$, i.e. σ' flows into σ.

Now we turn our attention to $\mathbb{R}^3$. In our attempt to compute $F(x)$ for a maximum x in $\mathbb{R}^3$, we mimic the setup that we built for $\mathbb{R}^2$.

Flow relation in $\mathbb{R}^3$. Let σ_1, σ_2 be two tetrahedra sharing a triangle t. We say $\sigma_1 < \sigma_2$ if σ_1 and its dual Voronoi vertex lie on the opposite sides of the plane of t.

It follows from the definition of $<$ that if $\sigma_1 < \sigma_2$, then the radius of the circumsphere of σ_1 is smaller than the ra-



(a) (b)

Figure 5: The closed stable manifolds of the maxima decompose the interior of the curve into four segments, the middle two segments are mergeable (a). The discretized version has six segments. The Gabriel edges (solid) among the Delaunay edges (dashed) form the boundaries of these segments. All four middle segments can be merged into a single segment.

dius of the circumsphere of σ_2. Thus, as in $\mathbb{R}^2$, the transitive closure $<^*$ is acyclic. For a maximum x let

$$\widetilde{F}(x) = \bigcup_{\sigma' < \sigma} \sigma' \text{ where } x \in \sigma.$$

So far everything seems analogous to the two dimensional case, but here we face two difficulties. First, Lemma 4 is no more valid, i.e. it may be that $F(x) \neq \widetilde{F}(x)$ for a maximum x. This is mainly because the stable manifolds of index 2 critical points may not be composed of Delaunay triangles. However, we could use $\widetilde{F}(x)$ as an approximation to $F(x)$. But, we face another difficulty. It might be that $\widetilde{F}(x)$ and $\widetilde{F}(x')$ are not disjoint for two maxima x and x'. The reason is that, for a tetrahedron σ, there may exist more than one tetrahedron σ' so that $\sigma < \sigma'$. This may lead σ to flow into two or more different maxima. However, it is interesting to notice the following.

Observation 4 *There exist no three tetrahedra $\sigma_1, \sigma_2, \sigma_3$ so that a tetrahedron σ satisfies $\sigma < \sigma_i$ for $i = 1, 2, 3$.*

In order to get pairwise disjoint sets $\widetilde{F}(x)$ we change the relation $<$ to a new relation $\tilde{<}$ so that for a tetrahedron σ there are no two tetrahedra σ_1, σ_2 with $\sigma \tilde{<} \sigma_1$ and $\sigma \tilde{<} \sigma_2$. Note that the *height* of a maximum x, i.e. its least squared distance to the sample points P, is the circumradius of the tetrahedron containing x. Define the *strength* of a tetrahedron σ as the largest of the heights of all maxima that it flows into.

Strengthened flow relation. We say $\sigma_1 \tilde{<} \sigma_2$ if

(1) $\sigma_1 < \sigma_2$ and

(2) there is no other tetrahedron σ_3 with $\sigma_1 < \sigma_3$ and the strength of σ_3 is larger than σ_2.

The transitive closure $\tilde{<}^*$ is acyclic since $<^*$ is. Now for a maximum x we redefine $\widetilde{F}(x)$ as

$$\widetilde{F}(x) = \bigcup_{\sigma' \tilde{<}^* \sigma} \sigma' \text{ where } x \in \sigma.$$

The sets $\widetilde{F}(x)$ are pairwise disjoint since no tetrahedron can flow into more than one maximum. We compute these sets as an initial segmentation of the shape represented by the finite sample P. Most of the segments but not all of them lie in the interior of the shape. One can obtain only inner segments after reconstructing the boundary of the shape from its sample using any of the known surface reconstruction algorithms.

We sort the maxima in decreasing order of their strengths and process them in this order. So, when we process a maximum x, all tetrahedra flowing into x and having a strength larger than that containing x have been claimed by some other maxima processed earlier. This is what is required by the definition of $\widetilde{F}(x)$.

STABLEMANIFOLD(P)
compute V_P and D_P;
determine the maxima Voronoi vertices;
sort the maxima in decreasing order of their heights;
for each maximum x in this order
  $\widetilde{F}(x) := \sigma$ where $x \in \sigma$;
  mark σ and all its triangles;
  while $\exists$ an unmarked triangle t in boundary $\widetilde{F}(x)$
    let $t = \sigma_1 \cap \sigma_2$ where $\sigma_2 \in \widetilde{F}(x)$;
    if σ_1 is unmarked and $\sigma_1 < \sigma_2$
      $\widetilde{F}(x) := \widetilde{F}(x) \cup \sigma_1$;
      mark σ_1 and all its triangles
    endif
  endwhile
endfor

Sometimes closed stable manifolds segment a shape unnecessarily into small features. For example, small perturbations in a shape can cause insignificant segmentation, see for example Figure 5(a). Also, at the discrete level, sampling artifacts may introduce even smaller segments, see Figure 5(b). We propose merging such small segments till two adjacent segments differ significantly.

For a shape $\Sigma \subseteq \mathbb{R}^d$, let $S(x)$ be a stable manifold of an index $d-1$ critical point x which belongs to the boundary of a closed stable manifold $F(y)$ for a maximum y. We say $F(y)$ is *shallow* with respect to $S(x)$ if the heights $h(x)$ and $h(y)$ are close to each other measured by a threshold ρ as $h(x)/h(y) \geq \rho$.

Mergeable stable manifolds. Two closed stable manifolds $F(x_1)$ and $F(x_2)$ are *mergeable* if they are shallow with respect to a shared stable manifold on their boundaries.

Merging all mergeable closed stable manifolds we obtain the feature segmentation of Σ. For example, the middle two segments for the curve in Figure 5(a) are mergeable. We can translate the definitions and hence the merging algorithm to the discrete setting easily. The height of a critical point x is measured with the circumradius of the lowest dimensional Delaunay simplex that contains x. This means, in $\mathbb{R}^2$, we merge two closed stable manifolds $F(x_1)$ and $F(x_2)$ if they share a saddle edge whose circumradius is more than $\rho < 1$ times the circumradii of the triangles containing x_1 and x_2.

In $\mathbb{R}^3$, we only compute approximations $\widetilde{F}(x)$ to a closed stable manifold $F(x)$ for a maximum x. Mimicking the definition and the algorithm in $\mathbb{R}^2$ we define mergeability of two approximated stable manifolds as follows.

ρ-mergeable stable manifolds. Let $\widetilde{F}(x_1)$ and $\widetilde{F}(x_2)$ be two approximated stable manifolds that share a triangle t. We say $\widetilde{F}(x_1)$, $\widetilde{F}(x_2)$ are ρ-mergeable if the circumradius of t is more than $\rho < 1$ times the circumradii of the tetrahedra containing x_1 and x_2.

The final algorithm to compute a feature segmentation of a shape $\Sigma \subseteq \mathbb{R}^3$ from a sample P is described below.

SEGMENT(P, ρ)
1 STABLEMANIFOLD(P);
2 Merge all ρ-mergeable segments;
3 Output the resulting decomposition.

Figure 6 shows some example segmentations in $\mathbb{R}^2$ and $\mathbb{R}^3$ respectively. The point samples for the 2D models are extracted from the boundary of 2D images. As a result they are quite noisy. Nevertheless, the models are segmented nicely. Our segmentation method can be applied to the point samples from molecular surfaces. The segmentation induces a decomposition of the surface which may be helpful in the molecular docking problem in drug designs [21].

5. Matching

For shape matching we take advantage of our segmentation scheme by matching two shapes with respect to their features. Given a point sample P of a shape Σ, we identify a small set of significant features from our feature segmentation. These features are then mapped to a set of weighted points called the *signature* of Σ. In order to measure the similarity of two shapes, we compare their signatures which boils down to matching two small weighted point sets.

Signature. Let $R_{P,\Sigma}$ denote the set of features that the function SEGMENT computes from a point sample P of a shape Σ. To simplify notations we use R_Σ for $R_{P,\Sigma}$. By definition a feature $r \in R_\Sigma$ is a collection of Delaunay triangles if Σ is a shape in two dimensions and it is a collection of Delaunay tetrahedra if Σ is a shape in three dimensions. For a Delaunay simplex σ let c_σ and v_σ denote the centroid and volume of σ, respectively. The *representative point* r^* of a feature r and its weight $\hat{r}$ are defined as

$$\begin{aligned} \hat{r} &= \Sigma_{\sigma \in r}\, v_\sigma \\ r^* &= \frac{\Sigma_{\sigma \in r}\,(c_\sigma \cdot v_\sigma)}{\hat{r}}. \end{aligned}$$

That is, the weight of r is its volume and its representative point is the weighted average of the centroids of all $\sigma \in r$, weight being the volume of each simplex. We call a feature r *significant* if its volume is more than a certain fraction of the total volume of the shape. Given a segmentation R_Σ of a shape Σ, the signature $\text{sign}(\Sigma)$ is defined as the set of weighted feature representative points, i.e.,

$$\text{sign}(\Sigma) \;=\; \{(r^*, \hat{r}) \mid r \in R_\Sigma \text{ is significant}\}.$$

The amount of similarity between two shapes is measured by first scaling them with bounding boxes and then scoring the similarity between their signatures. In order to score the similarity between two signatures $\text{sign}(\Sigma_1)$ and $\text{sign}(\Sigma_2)$, we need to align them first.

Let r^*, s^* be the representative points in $\text{sign}(\Sigma_1)$ and $\text{sign}(\Sigma_2)$, respectively, with maximum weights. We first translate $\text{sign}(\Sigma_2)$ so that r^*, s^* coincide. Then an alignment is obtained by rotating $\text{sign}(\Sigma_2)$ so that a line segment between s^* and another point of $\text{sign}(\Sigma_2)$ aligns with a line segment between r^* and another point in $\text{sign}(\Sigma_1)$. Certainly, there are $\Theta(mn)$ alignments possible where $|\text{sign}(\Sigma_1)| = m$ and $|\text{sign}(\Sigma_2)| = n$. Since m, n are typically small (less than ten), checking all alignments is not prohibitive.

For each alignment we compute a score based on the matching of weighted points. Both a similarity measure (positive) and a dissimilarity measure (negative) are taken

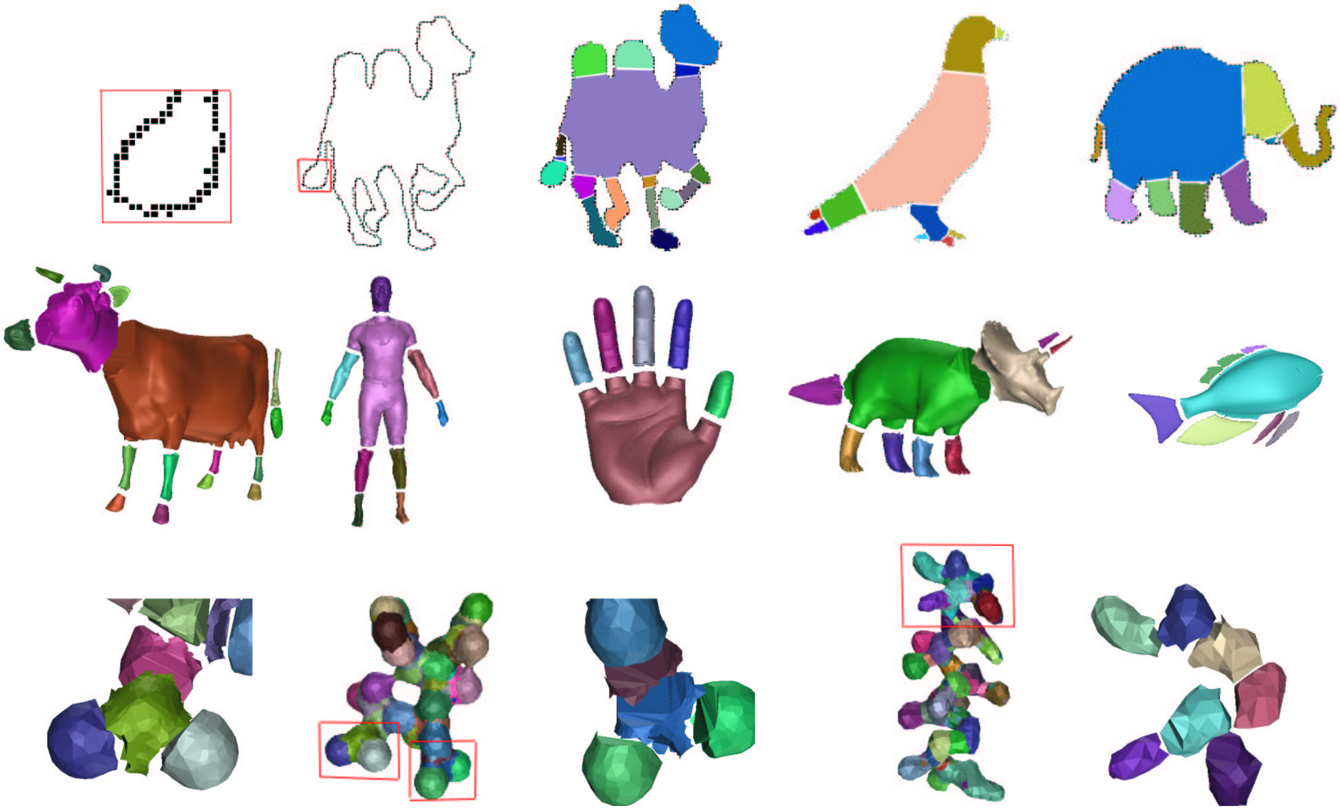


Figure 6: Segmentation of 2D and 3D models. In the leftmost picture of first row we zoom in the tail of the camel to show that the point sample is noisy as it is derived from the boundary extraction of a 2D image. The second row shows that the 3D models are segmented into so-called features. Third row shows how the segmentation can decompose molecular surfaces.

into account while computing the score. The maximum of all the scores is taken to be the amount of similarity and corresponding transformations give the best alignment.

Before we compute the score, the weights of the segments are normalized so that each weight is between 0 and 1. Next, for each point $q^* \in \text{sign}(\Sigma_2)$, we determine the Euclidean nearest neighbor, say p^*, in $\text{sign}(\Sigma_1)$. If $\|p^* - q^*\|$ is less than a *threshold*, we compute a *similarity score* as

$$1 - |\frac{\hat{p} - \hat{q}}{\hat{p} + \hat{q}}|$$

where the *threshold* is a parameter that tells how much tolerance we can have for the proximity of two features. The points in $\text{sign}(\Sigma_1)$ and $\text{sign}(\Sigma_2)$ that do not have nearest neighbors in the other set within threshold distance contribute to a *dissimilarity score* which is equal to the negative of their weights. Finally, we add both similarity and dissimilarity scores to obtain the score of matching between the two shapes Σ_1 and Σ_2.

In Figure 7 and 8 we show the result of our matching for shapes in two and three dimensions, respectively.

6 Discussions and Conclusions

Our results have shown that the segmentation is quite robust against small variations in shapes. For example, the three human bodies in the first row of Figure 8 are segmented similarly, namely into head, torso, two hands and two legs. Also the two hands in the second row are segmented similarly into five fingers and the palm. This is due to the fact that our approach emphasizes topology more than local geometry. Topological features in terms of the height function change relatively less with the local changes in geometry. There are a number of open questions remain to be addressed.

In some cases the segmented features deviate visibly from the intuitive ones. For example, the fingers in the hand in Figure 6 do not get separated from the palm where they meet. A small stump remains attached to the palm for each finger. We believe that we need a refined merging strategy

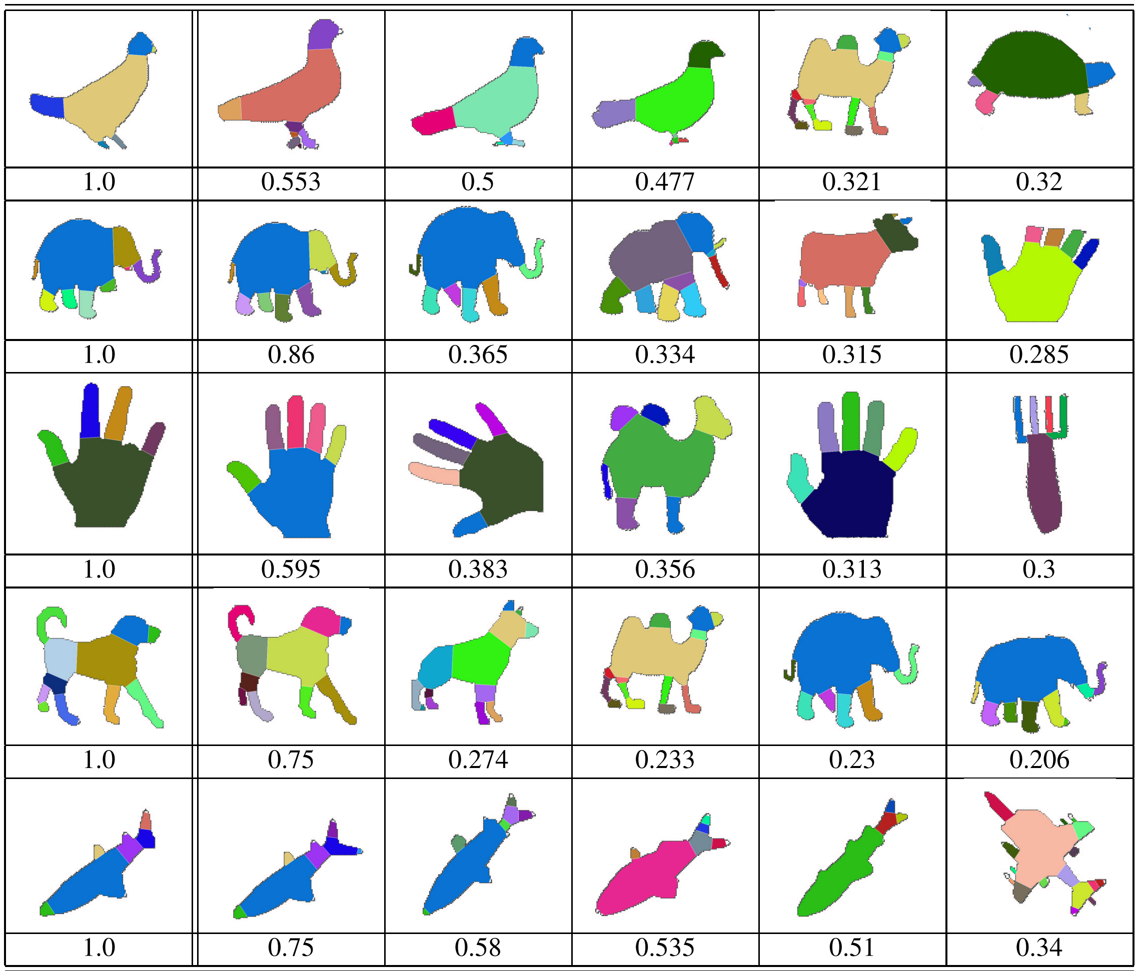


Figure 7: Matching results in 2D. Each row contains the matching scores for the query shape in the first column with the highest score 1.0.

to tackle this problem.

Although we mimicked the definition of features from the continuous space to the discrete domain, there is no quantitative estimate of the approximation. Specifically, can we claim that if the sampling density is beyond a threshold, then all defined features in the shape are approximated well?

In the shape matching we used the volumes of the segments as the weight of the representative points. In a sense, we took the volume of a feature to be its signature. Are there other measures that capture the signature of a feature more effectively? Currently we are investigating all these questions.

References

[1] H. Alt and L. J. Guibas. Discrete geometric shapes: matching, interpolation, and approximation: a survey. Tech. report B 96-11, EVL-1996-142, Institute of Computer Science, Freie Universität Berlin, 1996.

[2] K. Arbter, W. E. Snyder, H. Burkhardt and G. Hirzinger. Application of affine-invariant fourier descriptors to recognition of 3-d objects. *IEEE Trans. Pattern Analysis Machine Intelligence*, **12** (1990), 640–647.

[3] E. M. Arkin, L. P. Chew, D. P. Huttenlocher, K. Kedem and J. S. Mitchell. An efficiently computable metric for comparing polygonal shapes. *IEEE Trans. Pattern Analysis Machine Intelligence*, **13** (1991), 209–216.

[4] A. P. Askbrook, N. A. Thacker, P. I. Rockett and C. I. Brown. Robust recognition of scaled shapes using pairwise geometric histograms. *BMVC*, (1995), 503–512.

[5] R. Basri, L. Costa, D. Geiger and D. Jacobs. Determining the similarity of deformable shapes. *Vision Research* **38** (1998), 2365-2385.

[6] S. Belongie and J. Malik. Matching with shape contexts. *IEEE Trans. PAMI* **24** (2002), 509–522.

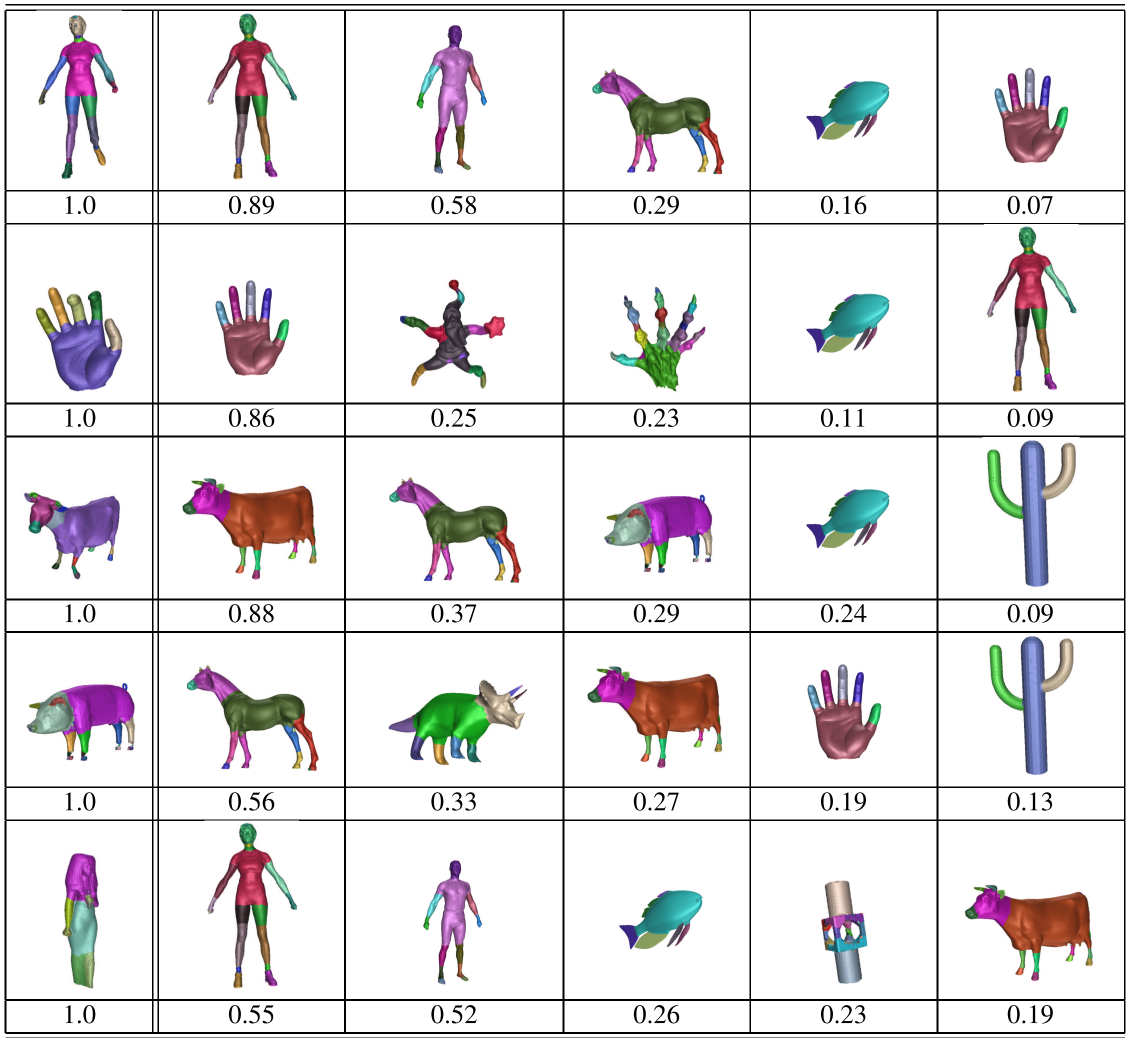


Figure 8: Matching results in 3D: each row contains the matching scores for the query shape in the first column with the highest score 1.0.

[7] P. Besl and R. C. Jain. Three-dimensional object recognition. *Computing Surveys*, **17** (1985), 75–145.

[8] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Trans. PAMI* **14** (1992) 239-256.

[9] T. A. Cass. Robust Affine Structure Matching for 3D Object Recognition. *IEEE Trans. PAMI* **20** (1998), 1265-1264.

[10] Y. Chen and G. Medioni. Object Modelling by Registration of Multiple Range Images. *Image abd Vision Computing* **10** (1992) 145-155.

[11] L. P. Chew, M. T. Goodrich, D. P. Huttenlocher, K. Kedem, J. M. Kleinberg and D. Kravets. Geometric Pattern Matching under Euclidean motion. *Comput. Geom. Theory Applications* **7** (1997) 113-124.

[12] H. Edelsbrunner. Surface reconstruction by wrapping finite point sets in space. *Ricky Pollack and Eli Goodman Festschrift*, eds. B. Aronov, S. Basu, J. Pach and M. Sharir, Springer-Verlag, to appear.

[13] H. Edelsbrunner, D. Letscher and A. Zomorodian. Topological persistence and simplification. *Proc. 41st IEEE Sympos. Found. Comput. Sci.* (2001), 454–463.

[14] H. Edelsbrunner, M. A. Facello and J. Liang. On the definition and the construction of pockets in macromolecules. *Discrete Apl. Math.* **88** (1998), 83–102.

[15] J. Giesen and M. John. The flow complex: a data structure for geometric modeling. *Proc. 14th Ann. ACM-SIAM Sympos. Discrete Algorithms*, 2002, to appear.

[16] S. K. Gupta, W. C. Regli and D. S. Nau. Manufacturing feature instance: which ones to recognize? *Proc. Sympos. Solid Modeling*, (1995), 141–152.

[17] M. Hilaga, Y. Shinagawa, T. Komura and T. Kunni. Topology matching for fully automatic similarity estimation of 3D shapes. *Proc. SIGGRAPH 2001*, (2001), 203-212.

[18] D.P.Huttelnocher, G.A.Klanderman and W.J.Rucklidge. Computing images using the Hausdorff distance. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **15** (1993), 850–863.

[19] C. E. Jacobs, A. Finkelsten and D. H. Salesin. Fast multiresolution image querying. *Proc. SIGGRAPH '95*, 277–286.

[20] A. E. Johnson and M. Hebert. Using spin-images for efficient multiple model recognition in cluttered 3-d scenes. *IEEE Trans. Pattern Analysis Machine Intelligence*, **21** (1999), 433–449.

[21] I. D. Kuntz. Structure-based strategies for drug design and discovery. *Science* **257** (1992), 1078–1082.

[22] F. Leymarie and B. Kimia. The shock scaffold for representing 3D shape. *Proc. 4th Internat. Workshop Visual Form.*, (2001), LNCS 2059, Springer-Verlag, 216–229.

[23] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition* **31** (1998) 983-1001.

[24] R. Osada, T. Funkhouser, B. Chazelle and D. Dobkin. Matching 3D Models with Shape Distribution. *Proc. Shape Modelling Int'l*, 2001.

[25] P. Petersen. Riemannian geometry. Graduate Texts in Mathematics 171. Springer (1997)

[26] R.J.Prokop and A.P.Reeves. A survey of moment based techniques for unoccluded object representation and recognition. *CVGIP:Graphics Models and Image Processing*,54(5):438-460,1992.

[27] Y. Rubner, C. Tomasi and L. Guibas. A metric for distributions with applications to image databases. *6th ICCV*, (1998), 59–66.

[28] S. Sclaroff and A. P. Pentland. Modal Matching for Correspondence and Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **17** (1995) 545–561.

[29] T. B. Sebastian, P. N. Klein and B. Kimia. Recognition of shapes by editing shock graphs. *Proc. ICCV*, (2001), 755-762.

[30] J. A. Sethian. Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science. Cambridge University Press, 2000.

[31] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson and S. W. Zucker. Shock graphs and shape matching. *Computer Vision*, (1998), 222–229.

[32] R. Sonthi, G. Kunjur and R. Gadh. Shape feature determination using the curvature region representation. *Proc. Sympos. Solid Modeling*, (1997), 285–296.

[33] G. Taubin and D. Cooper. Geometric invariance in computer vision, *Chapter Object Recognition Based on Moment (of algebraic) Invariants*, MIT press, 1992.

[34] R. C. Veltkamp and M. Hagedoorn. State-of-the-art in shape matching. Technical report UU-CS-1999-27, Utrecht University, the Netherlands, 1999.

[35] I. Weiss and M. Ray. Model-based recognition of 3D object from single vision. *IEEE Trans. PAMI*, **23** (2001), 116–128.

[36] D. Zhang and M. Hebert. Harmonic maps and their applications in surface matching. *CVPR '99*, 1999.