Principal component and Voronoi skeleton alternatives for curve reconstruction from noisy point sets

OSCAR RUIZ* , CARLOS VANEGAS, CARLOS CADAVID

EAFIT University, Colombia

Surface reconstruction from noisy point samples must take into consideration the stochastic nature of the sample. In other words, geometric algorithms reconstructing the surface or curve should not insist in following in a literal way each sampled point. Instead, they must interpret the sample as a "point cloud" and try to build the surface as passing through the best possible (in the statistical sense) geometric locus that represents the sample. This work presents two new methods to find a Piecewise Linear approximation from a Nyquist-compliant stochastic sampling of a quasi-planar C^1 curve $C(u) : R \to R^3$, whose velocity vector never vanishes. One of the methods articulates in an entirely new way Principal Component Analysis (statistical) and Voronoi-Delaunay (deterministic) approaches. It uses these two methods to calculate the best possible tape-shaped polygon covering the planarised point set, and then approximates the manifold by the medial axis of such a polygon. The other method applies Principal Component Analysis to find a direct Piecewise Linear approximation of $C(u)$. A complexity comparison of these two methods is presented along with a qualitative comparison with previously developed ones. It turns out that the method solely based on Principal Component Analysis is simpler and more robust for non self-intersecting curves. For self-intersecting curves the Voronoi-Delaunay based Medial Axis approach is more robust, at the price of higher computational complexity. An application is presented in Integration of meshes originated in range images of an art piece. Such an application reaches the point of complete reconstruction of a unified mesh.

Keywords: Curve Reconstruction; Surface Reconstruction; Unorganised Points; Range Imaging; Principal Component Analysis; Delaunay Triangulation; Voronoi Skeleton

1 Introduction

Reconstructing a curve or a surface from a point set is one of the most important problems in the reverse engineering of geometric models. In some cases curve reconstruction plays an important role in the surface reconstruction problem (Lee 2000). It is the goal of this paper to present two methods involving statistical (Principal Component Analysis -PCA) and deterministic techniques (Voronoi-Delaunay) for reconstructing a set of curves from noisy unorganised point sets. An application for surface reconstruction is presented, using data sets resulting from objects captured by range images. The references examined indicate that such a combination of methods has not been tried before for curve and surface reconstruction, or for range image mesh integration.

Even though this work will concentrate on quasi-planar curves, the statistical methods involved directly extend to arbitrary curves in $3D$. Two types of noisy unorganised point sets have been considered. One of them results from sampling and adding statistical noise to a set of mutually disjoint *regular* parametric curves (i.e. whose first derivative vector is continuous and never vanishes) $C_i(u)$ in R^3. The other point sample is originated in a cluster of individual meshes from range images. The point samples are assumed to comply with the Shannon or Nyquist criteria for digital sampling.

Problem Statement. Given a sample $S = \{p_0, \dots, p_N\}$ from an (unknown) set of mutually disjoint regular (open or closed) quasi-planar parametric curves $C_i(u)$ in R^3 and which may self-intersect, a PL (Piecewise Linear) estimation of each $C_i(u)$ is to be found. As seen later, without loss of generality we may assume that $C \subset R^2$.

The statistical methods which estimate the tangent to a curve $C_i(u)$ are not capable of determining by themselves the correct sense of the $\pm v$ tangent vector. For this reason we require that the curve has certain continuity in the

---

*Corresponding author. Email: oruiz@eafit.edu.co

derivative and that in the neighbourhood of each of its points it is well approximated by a straight line. That is, $C_i(u)$ must be C^1-continuous and its velocity vector must never vanish (i.e. the curve must be *regular*).

In this paper the stated problem is solved and an application of its solution is presented, for integration of range image meshes. To integrate a set of meshes of individual range images, the set of meshes is sliced by parallel planes. Each slice S_k turns out to be a coplanar set of points $S_k = \{P_{0_k}, P_{1_k}, \ldots, P_{N_k}\}$ with a strong statistical component stemming from the optical sampling error. The proposed algorithm finds a PL estimation of the curve $C_k(u)$ that adequately fits the points in the noisy unorganised point set S_k. The Literature Review section illustrates that such an integration of individual range meshes is still an open problem in several aspects. Section 5.1 discusses the application of PL curve reconstruction in detail.

Another application of the proposed algorithms in integration of individual range meshes arises when a particular slice k is missing or incomplete In the case of range imaging, this occurs when a portion of the object is not captured by any of the images. In such a case, point samples from levels $k-1$ and $k+1$ are projected onto the insufficiently sampled plane k. The resulting cross section on plane k must then be recovered from a possibly noisy point set. This point set should be treated with statistical tools, and the cross sections recovered should be the best fit to the planar point cloud contained in plane k.

A variant of the first type of noisy point sets (used to illustrate the Voronoi-Delaunay method) consists of a noisy sample of a self-intersecting planar parametric curve. Figure 1 shows a situation in which the local geometry of a planar slice (for example a Computer Axial Tomography - CAT) added to the presence of stochastic noise renders a set of points that look like the one in Figure 10(a). Clearly, less extreme situations may render an "8"-like section in the presence of a high level of stochastic noise. In the case of a sample of an "8"-like section two legal resulting PL approximations are equally likely: (a) *two* separate circular polygons, and (b) *one* polygon with a thin wasp waist. It is clear that near the self-intersecting point any algorithm may be confused. A survey of reverse engineering methods is presented in Varady, Martin and Cox (1997), being evident the use of curve reconstruction from point samples for generation of revolution or extrusion 2-manifolds. One of such applications is presented by Lee (2000). This application is particularly important in reverse engineering when the designer interactively tests the fitting of such surfaces to specific portions of the point set.
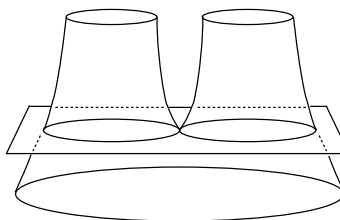

Figure 1. 2-manifold sample which renders a non-manifold curve.

2 Literature Review

Several solutions are available for curve reconstruction from point sets without noise. A survey on techniques for the case of closed, smooth, and uniformly sampled curves can be found in Edelsbrunner (1998). Methods for non-uniformly sampled smooth curves, and for uniformly sampled non-smooth curves are cited by Althaus *et al.* (2000). Some TSP (Travelling Salesman Problem) and tour improvement heuristics were used by Althaus and Mehlhorn (2000), and good experimental results were reported. In Amenta, Bern and Eppstein (1998) the PL approximation of a C^2 curve sampled in a dense pattern proportional to its local feature size (a modification of the Nyquist criterion) is discussed. Two graphs, the *crust* and β-*skeleton* are discussed, whose edge set exhaust the point sample. It should be noted that the curve reconstructed by these algorithms passes through each of the sampled points, and this type of solution is not adequate for the noisy point sets considered in the present paper.

The methods proposed for the case of non self-intersecting unorganised noisy point sets include spring energy minimization (Fang and Gossard 1992), implicit simplicial curves (Taubin and Ronfard 1996), α-shape polygonal boundaries and medial axes (Edelsbrunner and Mücke 1994), and moving least squares (Lee 2000). A review of these methods along with their limitations can be found in Lee (2000). Verbeek *et al.* (2001) approximate an open

curve by k segments that are least squares approximations of point subsets contained in Voronoi regions for sets of segment. By increasing k, better approximations to the curve $C(u)$ are found until a fitting criterion is met. However, the segments still need to be joined in a Hamiltonian graph, significantly adding to the complexity of the algorithm. The segments of the Hamiltonian graph may be larger than the segments found in fitting the point set. This has the effect of producing a PL approximation that may be deformed when compared with the $C(u)$ curve.

(Cheng *et al.* 2005) attack the problem of noisy point samples by computing a new point set having less noise as than the initial point set. The actual PL approximation to $C(u)$ is computed using a crust algorithm (in this case the NN Crust by Dey and Kumar (1999)). The new point set is calculated as follows: for each sample point p a thin rectangle is built with its main axis normal to the curve tangent and covering a certain number of point samples. The centre of such rectangle replaces p for the remaining of the algorithm run. The rectangle centres are closer to the $C(u)$ curve than the original sampled points. From all these rectangle centres one keeps the most external ones. In this way, the point set is pruned while a supporting width for crust algorithms is provided. At the end, a crust algorithm is called. In the method discussed in our paper, a ball $B(p, r)$ replaces the rectangle, and the centre of mass of the points inside the ball is assumed to be on $C(u)$. Also, the ball contains a point set whose main trend is tangent to $C(u)$ instead of normal to it. In our approach, no additional crust algorithm is needed, since the PL approximation to $C(u)$ is directly built using the centre of mass of those points in the sample which are contained in the ball.

Wang, Pottmann and Liu (2006) fit B-splines to a set of noisy point sets using curvature - based squared distance minimization. For this reason, the minimization requires the form of the equation (spline), and makes no attempt to attack noisy point sets with self - intersecting conditions. On the other hand, no discussion of the complexity of the algorithm is provided in time or in computational space. We feel that keeping the objective as a PL curve avoids the literal formulation of B-splines in the algorithm. Also, our research has as a goal the representation of non-manifold curve samples as PL non self-intersecting curves (i.e., manifold topologies), which allow for the subsequent usage of the PL curves in geometrical or topological constructs.

Kegl (1999) and Kegl and Krzyzak (2002) explore the recovery of a Principal Graph underlying a 2D point sample (e.g. a character meant to by pen strokes). The authors set up a numerical optimization algorithm that weights two competing criteria in the graph: (i) should as closely as possible follow the many pixels in the stroke, and (ii) should not have high curvature portions. An important feature for the application of this algorithm is that, since a character is sought, the final P.L. approximation does not have to be a manifold. Therefore, self-intersections are permitted (like in the "H" or "8" characters). In our case, the final result of the reconstruction should be a set of disjoint non self-intersecting curves, and therefore one must take care of higher requirements than the ones Kegl and Krzyzak (2002) and Kegl (1999) met.

Range Images and Point Set Surfaces Because the algorithms proposed in this paper are to be applied to the integration of range images, the authors consider that a review on range images is worth as a motivation for the reader. Range imaging offers a manner of digitizing the shape of three-dimensional objects. Because all opaque objects self occlude, no single range image suffices to describe the entire object, making necessary the combination of a collection of range images from diverse viewpoints into a single polygonal mesh that completely describes the object. Turk and Levoy (1994) create individual meshes for the different range images and clip them against each other for integration. Unfortunately, their integration method shows instabilities documented in Curless and Levoy (1996). Curless and Levoy (1996) integrate range images by creating a scalar field containing the minimal signed distance $f(x, y, z)$ from the point (x, y, z) to the object's surface. Afterwards, a Marching Cubes algorithm creates the B-Rep of the iso-surface $f(x, y, z) = 0$. A shortcoming of this method is the fact that the signed distance is calculated as a directional (instead of a scalar) property, and therefore there is no guarantee that the scalar field correctly registers the signed distance from a point to the surface. In Soucy and Laurendeau (1995) the very high computational cost of combining range image meshes after registration and surface meshing is discussed. In this reference overlapping components of the meshes corresponding to different range images must be identified, with a large computational cost, of the order $O(2^N - 1)$ where N is the number of range images. This reference unrealistically assumes the accuracy of the range data, as precision of the range data deteriorates in the periphery of each range image. In Zhou, Liu and Li (2006) a heuristic method for merging overlapping triangular meshes from range images is discussed. This article does not prove the correctness of the method exposed, which is based on the distance between triangles that are considered as overlapping. The less likely mesh is projected against the more likely one, based on a purely geometric projection, giving rise to topological inconsistencies that are not dealt with

rigorously.

For the direct treatment of the integrated point cloud from individual range images Hoppe et al (1992) use the k nearest point neighbours of a particular point p in the cloud to estimate the best local tangent plane. The plane is then used to construct the signed distance function $f(q) : R^3 \rightarrow R$ from point q to such plane. A Marching Cubes algorithm is then used to construct an approximation for the manifold $f(q) = 0$. This reference does not discuss the reconstruction of manifolds with border, nor the behaviour of the algorithm in incorrectly smoothing sharp edges of the piece. Indeed, their examples show a strong trend to filter out high frequencies. For these reasons, directly fitting surfaces to point sets has been an open research field since 1992. As a consequence, there has been a steady stream of publications in this direction. Ohtake et al (2005) use spherical influence regions to calculate most likely points on the surface and local normal vectors. For these authors and others, however, a difficulty with direct reconstruction of the manifold from the integrated point cloud remains in the fact that stitching together the local planes (triangles) gives rise to non-manifold topologies. Adamson and Alexa (2006) propose the computing of ellipsoidal weighting functions per sample to represent an implicit surface using supporting regions around each sample (Point Set Surfaces). It must be noted that such an approach does not explicitly compute the Boundary Representation of the model. Instead, it lends itself for visualization with ray casting.

The authors of the present article have found that the issues arising in curve reconstruction and in a possible application of it to range mesh integration are still an open problem in applied computational geometry. As seen from the literature review, curve reconstruction of self-intersecting curves is also unsolved. In range images, a reliable algorithm for mesh integration has not been proposed. Even in commercial systems (Geomagic 2006) such an integration requires the user interaction for correcting self-intersecting portions, holes, etc., that are left after the triangulation merges. Such facts have encouraged the authors to publish the present paper.

Section 3 examines the adaptation of statistical methods to be used in the present problem. Section 4 discusses the concepts necessary to implement the algorithms and their articulation in reaching the solution. Results for several types of point sets including non-smooth, self-intersecting, and non-uniform sets obtained with both methods are presented in Section 5. Section 5.1 describes an interesting integration of one of the methods to surface reconstruction from range images, and presents the results obtained for a real object. Section 6 discusses the computational complexity of the implemented methods. Finally, Section 7 draws the relevant conclusions, and proposes bases for future work.

3   Statistical Approach

The statistical approach for curve reconstruction from point samples has precursors in Hastie and Stuetzle (1989). In this reference, the authors define Principal Curves as smooth ones, which pass through the middle of, and are self-consistent with, a sampled cloud of n-dimensional data with dispersion (relative to the unknown curve) following a normal distribution (μ, σ).

3.1   *Principal Component Analysis (PCA)*

Although the following discussion treats noisy point sets in R^2 and R^3, it may be useful to know that the stochastic analysis presented is applicable to samples in n dimensions (in fact, the Principal Component Analysis method was developed for the treatment of samples in n-dimensional space, with $n >> 3$).

Let $S = \{p_i \in R^n : 1 \leqslant i \leqslant m\}$ be a set of m sample points in R^n. Without loss of generality one may assume that

$$\mu_1 = \mu_2 = ... = \mu_n = 0 \tag{1}$$

meaning that the expected value of the n-dimensional distribution or the p_i's is the origin of R^n. Let Σ be the covariance matrix of the sample, where $\Sigma_{i,j}$ is the covariance of the ith against the jth component of the p_i points.

One is interested in rotating S with an orthogonal transformation R such that the new set $S' = \{q_i \in R^n : 1 \leqslant i \leqslant m\}$ of transformed sample points $q_i = R * p_i$ presents maximal dispersion in the direction of the first axis of R^n, the second maximal dispersion in the direction of the second axis, and so on. For a $3D$ point set that has

a stochastic *linear* trend, establishing the direction of maximal dispersion is equivalent to identifying the direction vector of the line from which the sample was taken. For a $3D$ point set with an stochastic *planar* trend, establishing the direction of minimal dispersion identifies the normal vector of the plane from which the sample was taken.

Let X_p, Y_p, Z_p be the unit vectors pointing in the directions in which S has the largest (σ_x), second largest (σ_y) and smallest variance (σ_z), respectively. It may be shown that

(i) The pairs $(\pm X_p, \sigma_x)$, $(\pm Y_p, \sigma_y)$, and $(\pm Z_p, \sigma_z)$ are eigenvector - eigenvalue pairs of the matrix Σ:

$$\Sigma * (\pm X_p) = \sigma_x * (\pm X_p)$$
$$\Sigma * (\pm Y_p) = \sigma_y * (\pm Y_p)$$
$$\Sigma * (\pm Z_p) = \sigma_z * (\pm Z_p) \tag{2}$$

(ii) $\pm X_p$, $\pm Y_p$, $\pm Z_p$ are mutually orthogonal:

$$X_p \bullet Y_p = X_p \bullet Z_p = Z_p \bullet Y_p = 0 \tag{3}$$

(iii) $R * [X_p, Y_p, Z_p, O_p] = [X_w, Y_w, Z_w, O_w]$ and therefore:

$$R = \begin{bmatrix} X_p \ Y_p \ Z_p \ O_p \\ 0 \ \ 0 \ \ 0 \ \ 1 \end{bmatrix}^{-1} = \begin{bmatrix} X_p^T & 0 \\ Y_p^T & 0 \\ Z_p^T & 0 \\ O_p^T & 1 \end{bmatrix} \tag{4}$$

where $[X_w, Y_w, Z_w, O_w]$ is the World Coordinate System or a fixed reference frame. Without loss of generality, one may assume that $X_w = [1,0,0]^T, Y_w = [0,1,0]^T, Z_w = [0,0,1]^T, O_w = [0,0,0]^T$ and therefore the right hand side of item (iii) above is a clipped 4×4 identity matrix. Because an eigenvector can always be normalized, it can also be assumed that $|X_p| = |Y_p| = |Z_p| = 1$. Equation (4) results from the completion of the identity matrix in item (iii) and its (trivial) inversion.

As a result, $[X_p, Y_p, Z_p, O_p]$ is easily found and constitutes a right handed coordinate system. In particular, $[X_p, Y_p, Z_p]$ is an orthogonal matrix. As desired, a parametric line $p(\eta) = O_p + \eta * X_p$ which passes through the centre of gravity of the point cloud S is found by sorting and naming the eigenvector-eigenvalue pairs of Σ so that $\sigma_x \geq \sigma_y \geq \sigma_z$.

From Equations (2) and (4) it is clear that for quasi-planar data set, the eigenvector Z_p associated to σ_z is the estimation of the direction normal to the fitting plane, since σ_z is by definition the direction of minimal dispersion of the (quasi-planar) set of points. Conversely, for line data, the estimation of the direction vector of the line is the eigenvector X_p, since it is associated to the eigenvalue σ_x representing the maximal dispersion.

3.2   *Least Squares Fitting*

Section 3.1 explained how the coordinate system $[X_p, Y_p, Z_p, O_p]$ is calculated using PCA, by computing the eigenvector-eigenvalue pairs $(\pm X_p, \sigma_x)$, $(\pm Y_p, \sigma_y)$, and $(\pm Z_p, \sigma_z)$, of matrix Σ . Because geometric kernels do not usually have routines for calculation of n-dimensional eigenpairs, a method was devised for the 3-dimensional case at hand. The method takes advantage of the fact that point samples from Coordinate Measurement Machines, Machine Tool stylos, CAT scans, etc., are planar or quasi-planar. As a consequence, a very close estimation of the lowest dispersion direction (the vector Z_p normal to the plane) can be easily achieved. The point cloud projected on this plane loses one dimension and therefore the problem becomes 2-dimensional. Therefore, a solution of the eigenpair problem in Equation (2) can be achieved as an extension of a Least Squares (LS) fitting. The LS method cannot be directly applied since it is based on the *implicit* equation $y = mx + b$, which has no solution if m is the

tangent of $\pm 90^o$. A random rotation around z, followed by LS fitting and the corresponding counter rotation of the point data set, avoids this problem and allows to express the 3D trend of the point cloud in terms of a *parametric equation* $p(\eta) = O_p + \eta * X_p$.

In two dimensions, the LS method detects the trend m of a linear phenomenon. Since the 3D problem at hand is projected into 2D space, finding m in 2D reduces to calculating the projection of the 3D direction vector X_p of $p(\eta)$ onto the best fitting plane for the point set. Notice that the point set is not exactly planar because of the machine tool sampling errors. Since Least Squares is just a PCA in two dimensions, in what follows, "PCA" and "Least Squares" should be read as synonyms.

3.3 *Point Sample Partition*

Regardless of the method employed to estimate a PL approximation for the curves, it is capital to recognize the fact that the data set must be partitioned into the data sets originated from the individual curves $C_i(u)$. In order to perform such a partition let us define an equivalence relation on the point set S, as follows. If the sampling conditions are anisotropic and constant over R^3, a point $p \in S$ is said to "be the extended neighbour" of a point $q \in S$, if and only if there exists a sequence of points of the sample S starting at p and ending at q such that no two consecutive points of the sequence are farther apart by more than a fixed distance ϵ from each other. Let $r(p,q)$ be read as "p is an extended neighbour of q". Formally, two points p, q are Extended Neighbours of each other, whenever there exists a sequence $[q_1, ..., q_w]$ such that each $q_i \in S$, $q_1 = p$, $q_w = q$ and $|q_i - q_{i+1}| \leq \epsilon$. The $r(\ )$ relation defined above is an equivalence relation since it satisfies:

(i) $r(P_i, P_i)$ (reflexive: a point P_i is extended neighbour of itself).

(ii) $r(P_i, P_j) \wedge r(P_j, P_k) \rightarrow r(P_i, P_k)$ (transitive: if P_i and P_j, and P_j and P_k are extended neighbours, P_i and P_k are so).

(iii) $r(P_i, P_j) \rightarrow r(P_j, P_i)$ (symmetric: if P_i is extended neighbour of P_j then P_j is extended neighbour of P_i).

This equivalence relation $r(\ )$ splits S into subsets $S_1, S_2, ...$ with the property that $r(P_i, P_j)$ holds (are extended neighbours) if and only if P_i and P_j belong to the same S_k. Properties (i), (ii) and (iii) of the relation $r(\ )$ imply that $\cup_i S_i = S$ and $S_i \cap S_j = \phi, i \neq j$. Each S_i of the partition happens to be the set of points sampled from the curve $C_i(u)$. The partition of the set S by the equivalence relation $r(\ )$ is realized by using a standard algorithm of *transitive closure* which will not be discussed here.

4   Algorithms

Two algorithms for determining a PL approximation for quasi-planar 1-manifolds in R^3 are presented in this section, along with two figures that show partial results obtained at the main steps of each one of them.

4.1 *Data Pre-Processing*

The point data must be pre-processed in the following sequence: (i) Scaling: to guarantee that a standard bounding box of the set S is available (PCA requires such a box). (ii) Partition: to divide S into subsets, each one containing the points of S corresponding to an individual $C_i(u)$ curve. (iii) Identification of Best Plane: to find a statistical plane Π fitting the quasi-planar point set S. (iv) Correction to Planar Set: to project S onto Π in order to have a perfectly planar point set. (iv) Transformation to XY Plane: to use the algorithmic results in literature which deal with point sets in the XY plane. Step (ii) is required since several $C_i(u)$ curves may have been sampled and the point set would represent several unrelated curves. In what follows, the notation $C_i(u)$ will be changed to $C(u)$ since the analysis is per curve. A post-processing step consisting in reversing the transformations performed in the pre-process, is necessary in order to bring the found solution back to the original space.

4.2   *Curve Reconstruction with Least Squares*

After the data pre-processing steps mentioned in Section 4.1 take place, the Least-Squares-based algorithm takes as input a quasi-planar set S, and returns a polyline that fits these points by performing the steps discussed below and displayed in Figure 2.
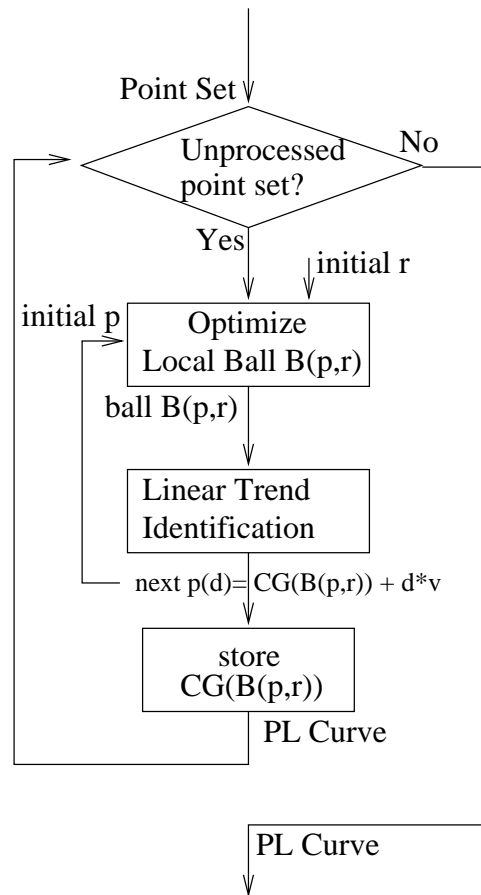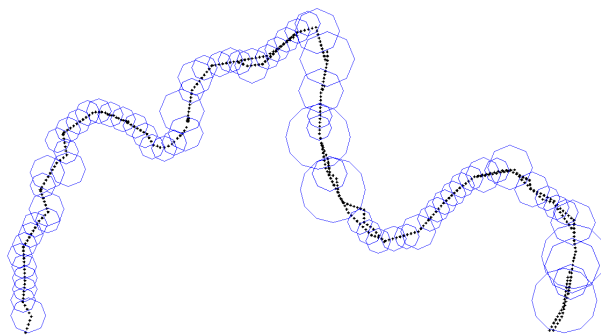

Figure 2.   Curve Reconstruction with Principal Component.

4.2.1   *Optimal Local Point Set Estimation.* Given a noisy unorganised point set, resulting from a stochastic sample with variance $[\sigma_x, \sigma_y, \sigma_z]$ of a planar 1-manifold $C(u)$ (possibly open) lying on plane Π in R^3, one is interested in estimating the tangent line $dC(u)/du|_{u=u^*}$, at point $C(u^*)$ of the curve $C(u)$. PCA and Least Squares are applied to points of the sample which are contained inside a ball $B(P_s, R)$, centred at a seed point P_s and having radius R. Two competing aspects must be compromised: (i) the ball should be small enough so that data set S can be considered to fit a linear estimation of the local tangent; (ii) the ball should be large enough so that the goodness of the linear estimation is kept. To achieve (i) and (ii) an iterative search is conducted for a combination of P_s and R, optimal for the linear fitting of local neighbourhoods of S. The iterative search starts with a ball $B(p(0), r(0))$ enclosing a set $S(0)$ of points. Let $\epsilon(p, r)$ be a function that associates a least-square regression fitting error to the points inside a ball with centre p and radius r. It is desired to find the values of p and r that minimize ϵ. Applying the PCA to the point set, a measurement of the fitting error is found. In the $k - th$ iteration a new value ot $r(k)$ is proposed ($r(k+1)$), which changes the size of the ball $B(r(k+1), p(k))$. This ball, in turn will enclose a different set of points $S(k+1)$, with new centre of gravity $p(k+1)$. The fitting of a new straight line to the set $S(k+1)$ will render a new fitting error. The iterations stop when such an error has a local minimum. This *ad hoc* process was found to have good convergence.

4.2.2 *Piecewise Linear Reconstruction of $C(u)$.* In the following discussion the term $B(p, r)$ will mean both the ball with radius r centred at p, and the subset of the point sample contained in such a ball. The context will define which meaning is intended. The algorithm in Figure 2 performs an estimation of the linear trend of the points in the optimized ball $B(p, r)$. Such an estimation produces a parametric equation for a straight line in space $p(\eta) = O_p + \eta * v$ with $|v| = 1$, where O_p is the centre of gravity of the points inside $B(p, r)$, v is the linear trend of the line (also called X_p in section 3.1). O_p is denoted as $CG(B(p, r))$ in Figure 2. Such a point is stored directly in the linear estimation of the $C(u)$ curve. The ball for the next iteration is initially centred at $O_p + d * v$, where d is the progression step of the algorithm and $|v| = 1$. Since $B(p_i, r_i)$ and $B(p_{i+1}, r_{i+1})$ intersect, it is clear that each sampled point may be used in several balls, and therefore in the estimation of successive tangents. Notice that the index i corresponds to already optimised balls in evolving localities of the curve $C(u)$ such that $|p_{i+1} - p_i| \approx d$. In Figure 2 the need for determining whether $+v$ or $-v$ is the correct trend is omitted (recall that PCA returns $\pm v$). This is easily done by ensuring that $v_i \bullet v_{i+1} > 0$. The later requirement is reasonable since the curve $C(u)$ is assumed to be regular.

The algorithm will continue as long as there are enough available points of the set S (see section 3.3) which fall inside a ball. Each point can be used in several balls, being their number set by the user. In Figure 2 the marking of the multiply used points of S is omitted for the sake of clarity. When this algorithm terminates, the curve $C(u)$ has been piecewise linearly estimated.

A noisy point set generated from a range image Multi-Mesh sample is shown in figure 3(a), together with the balls used by the reconstruction algorithm. Figure 3(b) shows the resulting reconstructed curve.

(a) Noisy Point Set from Range Image Multi-Mesh sample and Balls used in the reconstruction process.
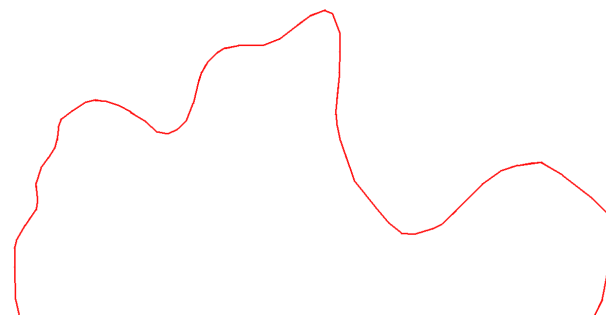
(b) Reconstructed Curve.

Figure 3. PCA-based Reconstruction.

4.3 *Principal Curve via Delaunay Triangulation*

The following discussion will be illustrated using a planar 1-manifold with border (open $C(u)$). Later on, the concepts explained will be applied on self-intersecting (i.e. non-manifold) planar curves.

For planar self-intersecting curves, PCA alone is not robust enough. Additional processing is required since the points in the neighbourhood of the self-intersection are exhausted for purposes of PCA estimation as the PL approximation crosses the first time over the intersection neighbourhood. As the PL curve revisits the intersection neighbourhoods no points are available for identifying the trend of the curve, and the algorithm tends to look for another point (i.e. curve) neighbourhood where to work, without really having reproduced the intersection. The result is an incomplete curve stage, therefore missing the self-intersection detail.

To deal with self-intersecting curves, it was decided to determine the tape-shaped polygon T_σ covering S (definition below). Figure 4 displays the algorithm discussed next.

Definition. Tape-shaped Polygon T_σ. Let $C : R \to R^3$ be a planar regular parametric curve, which may self-intersect. Without loss of generality let us assume that $C \subset R^2$.

Given $\sigma > 0$ a real number, define $T_\sigma = \{p \in R^2 : d(p, C) \leq \sigma\}$. There exists $\sigma_0 > 0$ such that if $0 < \sigma \leq \sigma_0$ then for every $p \in T_\sigma$ (i) the set of points $\{q_{p,1}, ..., q_{p,r_p}\} \subset C$ formed by those points whose distance to p equals $d(p, C)$ is finite. The $q_{p,i}$ points in C are the ones which realise the distance from p to C; (ii) the distance between any two points in the set $\{q_{p,1}, ..., q_{p,r_p}\}$ is less than 2σ. Observe that σ is dictated by the precision of the

Point Set

Delaunay
Triangulation

Delaunay
Triangles

Triangle
Filtering

Tape−shaped   L={L0,L1,...Lm}
Polygon

Medial Axis
Approximation

Medial Axis
Graph

Graph
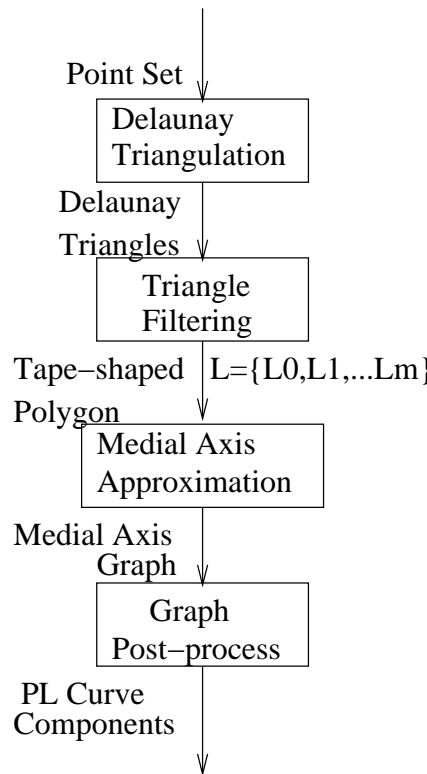Post−process

PL Curve
Components

Figure 4. Line Reconstruction through Delaunay-Voronoi Techniques.

measurement device which samples C. We assume that the measurement device allows a precision of $\sigma \leq \sigma_0$ and therefore that $S \subset T_\sigma$. Note that the $q_{p,i}$ could be regarded as the points to be sampled in the curve C in absence of sampling noise and that if C is non self-intersecting then q is unique for each $p \in S$.

For small enough values of σ_0 (Nyquist samples) T_σ resembles a tape region covering the curve C. Let us define $Q_S = \{q \in C : d(p,q) = d(p,C), \text{for some } p \in S\}$. Note that if S has no statistical noise, $Q_S = S$ and Q_S would be a noise-free Nyquist sample of C.

4.4 *Approximation of T_σ*

Under the condition of S being a Nyquist-compliant sample, this article proposes an algorithm to approximate the tape-shaped polygon T_σ. The algorithm follows three steps:

(i) calculates the Delaunay Triangulation of S, $DT(S)$;
(ii) then selects from $DT(S)$ small triangles;
(iii) and finally, makes T_σ the boundary of the union of the triangles selected in (ii).

In order to apply such a method, an estimation of what a "small triangle" is, should be made precise. For this purpose the typical area and edge length of Delaunay triangles belonging to T_σ need to be estimated. To do that, PCA is iteratively run on neighbourhoods of the data set, thus determining the line $p(\eta) = P_0 + \eta * v$ that best approximates the tangent to the $C(u)$ curve in that neighbourhood. The points of S that produce such a fit are contained inside a ball $f_D * B(P_0, R_0)$ approximately centred on a local neighbourhood of $C(u)$. Delaunay triangles contained within a scaled version of this ball, namely $f_D * B(P_0, R_0)$ (with $f_D = 1.3$ being an empirically chosen enlarging factor) might be considered as "typical" of the ones forming T_σ, rendering "typical area" $\bar{A}$ and "typical edge length" $\bar{l}$ values.

One considers that a triangle is small if either of the following criteria (Fortune (1992) and Guibas and Stolfi (1985)) holds:

(i) **Enclosure**: Accept a Delaunay triangle DT_i if it is contained within the local PCA ball, that is, if $DT_i \subseteq B(P_0, R_0)$ where $B(P_0, R_0)$ is the best local PCA ball (see Figure 5(c)).

(ii) **Area and Edge Length**: Accept a Delaunay triangle DT_i if its Area or maximal Edge Length are small. That is, if $Area(DT_i) \leq f_A * \bar{A}$ or if $E_{max} \leq f_l * \bar{l}$, respectively, for fixed constants f_A and f_l.

We give an informal discussion for the correctness of the procedure to obtain an approximation of T_σ. The tests run gave a good performance in the filtering of Delaunay triangles. An advantage of the implemented algorithm is that the application of PCA to the local neighbourhoods of the point cloud allows the estimation of the sizes of the triangles to be deleted and to be kept.

Let us suppose that, contrary to the assumption, a large triangle $DT_i = [v_j, v_k, v_l]$ belongs to T_σ. Since it is a Delaunay triangle, its circumcircle contains no points of S. But since DT_i is a large part of T_σ, a large portion of T_σ contains no sample points, contradicting the fact that S is a Nyquist sample. On the other hand, suppose that a small triangle $DT_i = [v_j, v_k, v_l]$ is not entirely contained in T_σ. If DT_i is completely outside T_σ, then it creates a contradiction since $S \subset T_\sigma$. If v_j, v_k, v_l are in T_σ but the triangle joins two approaching branches of C, the sample S is characteristic of a non-manifold situation and therefore DT_i is part of T_σ.

For the sake of simplicity T_σ will be denoted simply by T. An *approximation* of the medial axis of T, called here the *skeleton of T*, is the sought PL approximation of the $C(u)$ curve. Since the skeleton is a graph, it needs to be post-processed in order to extract from it the PL approximation of $C(u)$.

Figure 5(a) shows a data set from a planar non self-intersecting curve sampled stochastically. This figure presents a data set which has been already resized, its best plane estimated, and their points projected onto this plane, which produces a planar set. The Delaunay Triangulation of this point set is displayed in Figure 5(b).

4.4.1 *Polygon Synthesis based on Filtered Delaunay Triangulation.* The polygon L_0 obtained after application of criteria (i) and (ii) is shown in Figure 5(d). Observe that L_0 has no holes for this example. In that figure light triangles are the accepted ones based on the PCA criterion and dark triangles are the ones accepted based on area or edge length criteria. The following relations hold among accepted Delaunay triangles and their edges (Mantyla 1988) :

(i) Each edge of an accepted Delaunay triangle DT_i has one or two accepted triangles incident to it.
(ii) Edges $e_{i,j}$ in which Delaunay triangles DT_i and DT_j are incident are internal to the tape-shaped region T.
(iii) Edges e_i in which only one Delaunay triangle DT_i is incident form the boundary ∂T. They may be either in the outermost or in an internal loop.

### 4.5 *Medial Axis VS. Principal Curve*

Figure 5(d) presents the minimal polygon T that covers the point set S. Its border ∂T, built by filtering the original Delaunay Triangulation, is coloured black in Figure 5(e). A very fine resample of the border ∂T) is then performed, and a Delaunay triangulation for this new point set is calculated. This new Delaunay triangulation also appears in Figure 5(e).

An approximation to the *medial axis* $MA(T)$ of T is a skeleton $SK(T)$, which is built in the following manner (Geiger (1993), Boissonnat (1988), Ogniewicz (1994)):

(i) Construct the Voronoi Diagram $VD(T)$ and Delaunay Triangulation $DT(T)$ of the vertices of T (see Figure 5(e)).
(ii) Keep from $DT(T)$ only those Delaunay triangles contained in T. Call this set $\overline{DT(T)}$.
(iii) Keep from $VD(T)$ only those Voronoi edges which are finite and are dual to the edges in $\overline{DT(T)}$. Call this set $\overline{VD(T)}$.
(iv) If $\overline{VD(T)} \nsubseteq T$ then re-sample ∂T with a smaller interval and go to step (i) above. Otherwise, $\overline{VD(T)}$ is the sought skeleton of T, $SK(T)$.

As it is evident from Figure 5(f), the skeleton $SK(T)$ of the polygon T is a PL approximation of the 1-manifold $C(u)$.

Notice that several resamples of ∂T may be needed in order to converge to $SK(T)$. Figure 5(e) shows one such resample. The boundary ∂T of the S-shaped polygon T in Figure 5(f) is sampled with a small enough interval. This tight sampling guarantees that the portion of the Voronoi Diagram confined to T, $SK(T)$, is acceptable as an
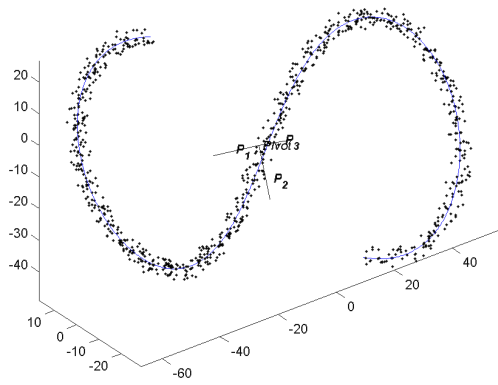
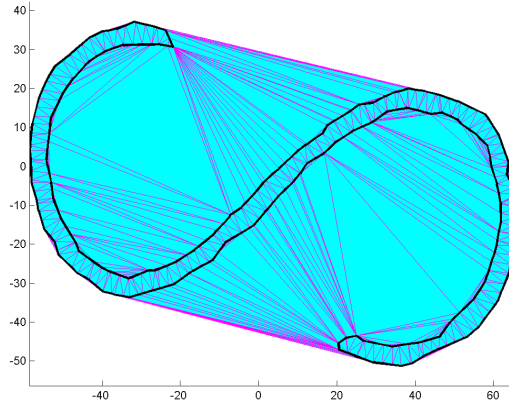(a) Point Sample of Planar S-shaped $C(u)$ Manifold.
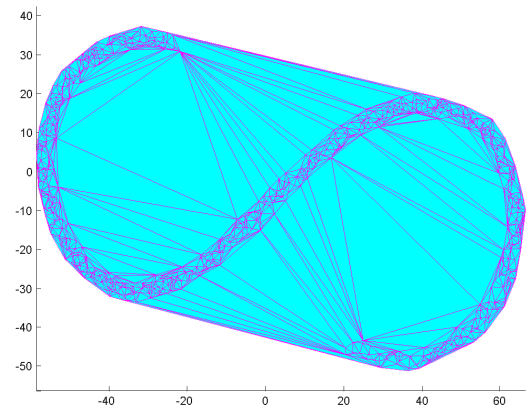
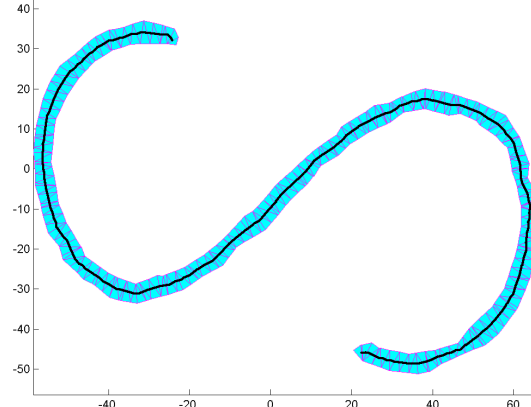(b) Delaunay Triangulation of S-shaped Planar Point Sample.
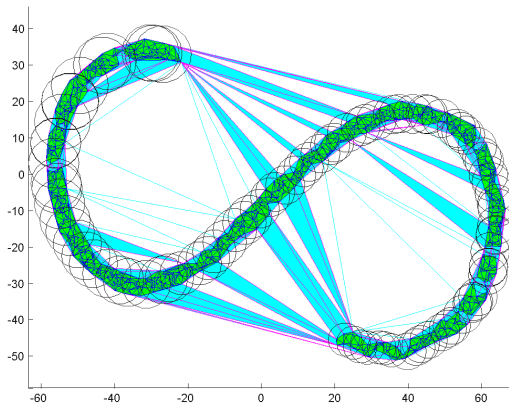
(c) Filtering of Delaunay Triangulation with PCA Balls.
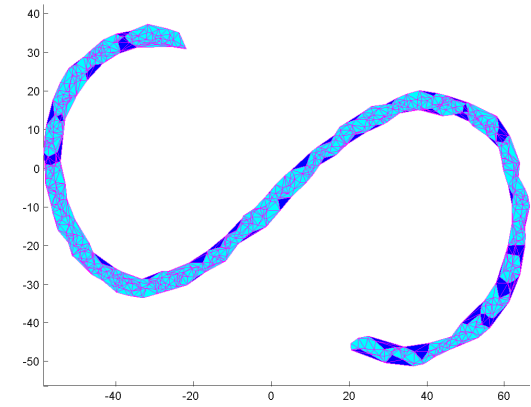
(d) Triangles Selected by Area and Length Criteria.

(e) Tape Polygon and its Delaunay Triangulation.

(f) Filtered DT and Skeleton.

Figure 5. Piecewise Linear Approximation of S-shaped $C(u)$ by Combined PCA and Voronoi-Delaunay Methods.

approximation of $MA(T)$, the medial axis of T.

5 Results

Section 5.1 illustrates three PCA curve reconstructions obtained for diverse point sets. It also discusses the application of PCA-based curve reconstruction to surface reconstruction from range images. Section 5.2 illustrates the results obtained using the Delaunay Triangulations methodology in dealing with the PL Approximation of planar
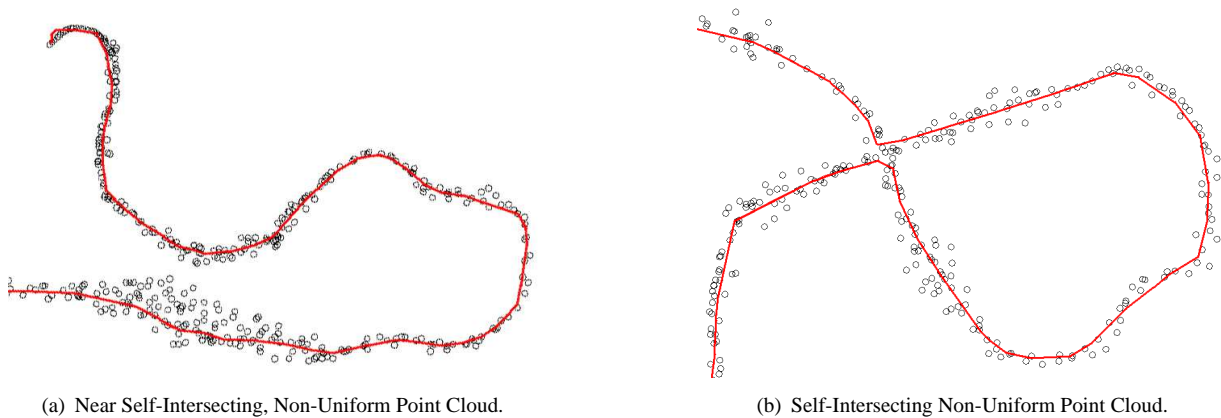
(a) Near Self-Intersecting, Non-Uniform Point Cloud.   (b) Self-Intersecting Non-Uniform Point Cloud.

Figure 6. Curve reconstructions obtained for different point sets by Least-Squares-based process.

1-manifolds without Border (closed $C(u)$).

5.1   *Least Squares Fitting Results*

The PCA-based algorithm was tested on several noisy unorganised point sets, which include non-uniform, non-smooth, near self-intersecting, and self-intersecting ones. Figures 6(a) and 6(b) present the results obtained for two sets, each one having some of these features. Near self-intersecting, non-uniform point clouds, as the one shown in Figure 6(a), can be adequately reconstructed by varying the length of the segments of the reconstructed polyline, considering the dispersion of points contained in each ball. The radius optimization process, described in section 4.2.1, turns out to be useful for this purpose.

In Figure 6(b) a point set sampling a self-intersecting curve $C(u)$ is displayed. As mentioned in Section 4.3, a PCA algorithm alone is not robust enough for reconstructing self-intersecting point clouds. However, due to the randomness of the starting point of the reconstruction mentioned in Section 4.2.1, certain runs can result in adequately reconstructing the PL approximation of $C(u)$, while other runs will not. Because of this, the skeleton method for curve reconstruction was considered.

Notice that criteria for identifying the ends of *open* noisy point sets are needed in order to *correctly* reconstruct open curves. These criteria include the fact that when the PCA algorithm finds an end of the curve $C(u)$, the evolution to a next centre of the fitting ball $B(p, r)$ is possible only in one direction. This condition allows to discriminate samples of open vs. closed curves. In the example discussed, (Aphrodite data set), however, all the sampled curves are closed.

5.1.1   *Application to Surface Reconstruction from Range Images.*   Range imaging is a technique for digitizing three-dimensional objects, given a set of range images. A range image is a function $I \times J \to R^3, \langle i, j \rangle \mapsto P_{ij}$, where $I \times J$ is the grid of pixels in the range image, and $P_{ij} = \langle x_{ij}, y_{ij}, z_{ij} \rangle$ is the point in the surface of the optically sampled object, captured by the pixel in position $\langle i, j \rangle$ of the grid of pixels.

As no single range image suffices to describe the entire object, it is necessary to combine a collection of range images (see Figs. 7(a) and 7(b)) into a single triangular mesh that completely describes the object. The steps listed below were followed in order to generate such mesh from the individual pictures (considered already registered with respect to each other): (i) Construction of the individual mesh M_i for each individual range image R_i (Figs. 7(a) and 7(b)) ; (ii) slicing of the complete set of meshes $M_i, i = 1, 2, ...$ with a set of parallel, equi-spaced planes, thus building planar samples of points; (iii) reconstruction of a set of curves (contours) from the sampled points by using the algorithm discussed in Section 4.2 (see contours in Figure 8); and (iv) use of an algorithm for surface reconstruction from planar slices. In this case, the algorithm discussed in Ruiz *et al.* (2005) was used. The reconstruction of Aphrodite's head is presented in order to illustrate the mesh integration process. The range images used were a courtesy of Fraunhofer Inst. for Computer Graphics, Darmstadt, Germany.

In step (ii), a set of parallel planes are defined, and the intersection between each plane and the collection of shells

(a) $i - th$ Mesh from Front Range Image i Aphrodite.

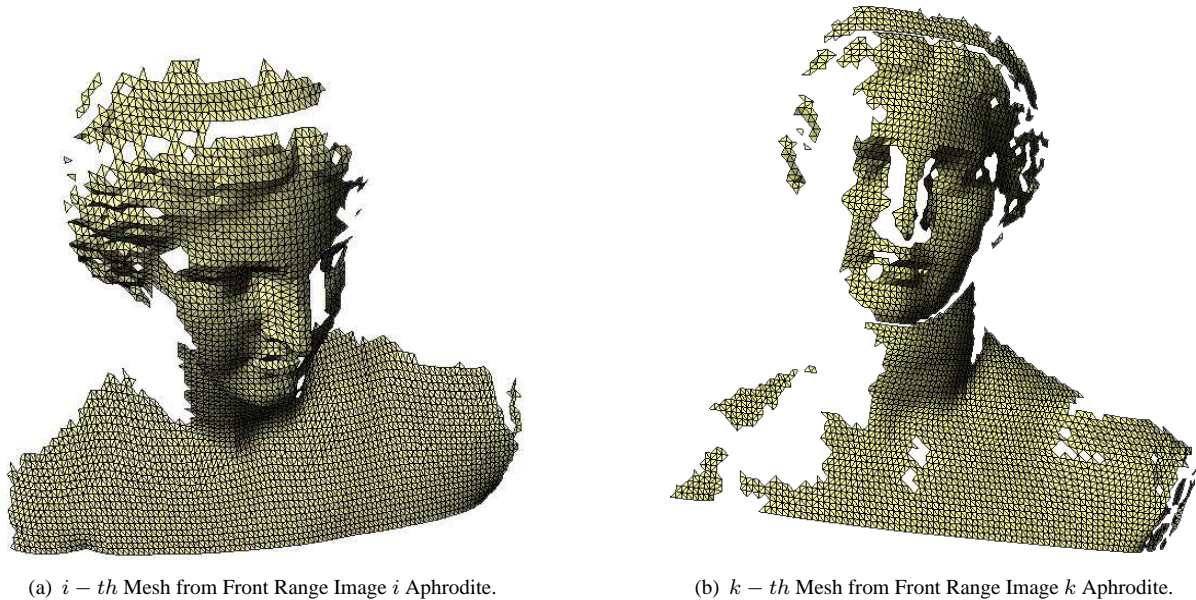(b) $k - th$ Mesh from Front Range Image k Aphrodite.

Figure 7. Range Image Data Set. Courtesy from Fraunhofer Inst. Computer Graphics, Darmstadt, Germany.

recovered from the range images is calculated. A set of planar samples of points $S_1, S_2, \ldots, S_k, \ldots$ is generated by sampling the polylines resulting from each intersection. Figure 3(a) shows one such coplanar sample $S_k = \{P_{0_k}, \ldots, P_{N_k}\}$ for Aphrodite's head model.
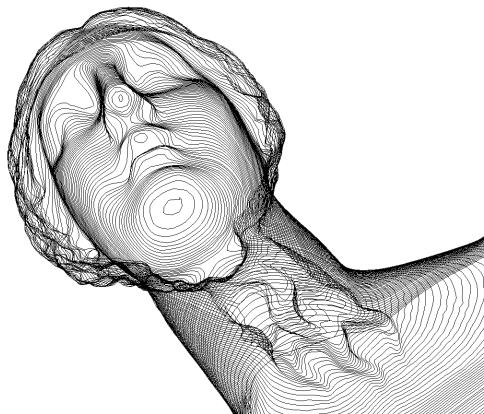

Figure 8.   Aphrodite's head contours recovered from planar samples of points. Test data courtesy from Fraunhofer Inst. for Computer Graphics, Darmstadt, Germany.

More than 100 levels (the number and separation dictated by the Nyquist criterion applied in the axial direction) of slicing were obtained from sampling the collection of meshes corresponding to Aphrodite's sculpture head and neck, and the same number of polylines were reconstructed from these sets (Figure 8). In spite of the large number of range images available for Aphrodite's sculpture, some of its regions were not covered by any of these, and therefore several sets of points needed to be manually completed. Once the sets were completed, none of the reconstructed polylines were edited. The surface reconstructed from the integrated, stochastically recovered contours is shown in Figures 9(a) to 9(c). Figures 9(a) and 9(b) correspond to resampling planes which are not orthogonal, and to an unfinished reconstruction (there is still a border). Figure 9(c) represents the integrated result for slicing planes parallel to plane XY. The final Aphrodite's surface reconstruction is shown in figure 9(d).

### 5.2   *Medial-Axis, Delaunay Triangulation Results*

Application of Medial Axis or Delaunay Triangulation methods is justified when the sampled curve $C(u)$ is self-intersecting. For this reason, these methods were not tested with the Aphrodite data set, but with planar self-
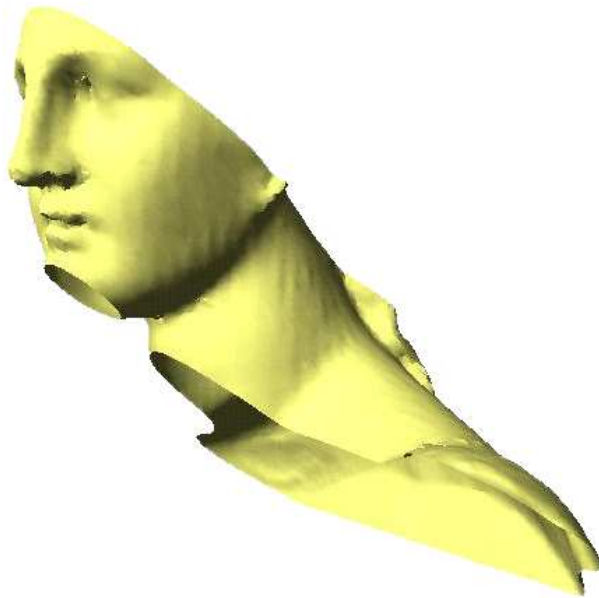
(a) Integrated Aphrodite with border. Smooth Render.
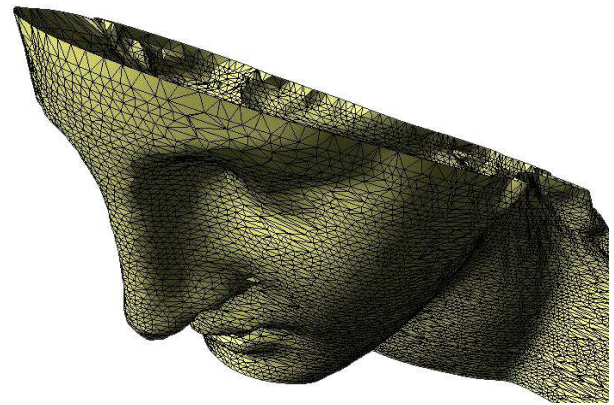
(b) Integrated Aphrodite with border. Wireframe.
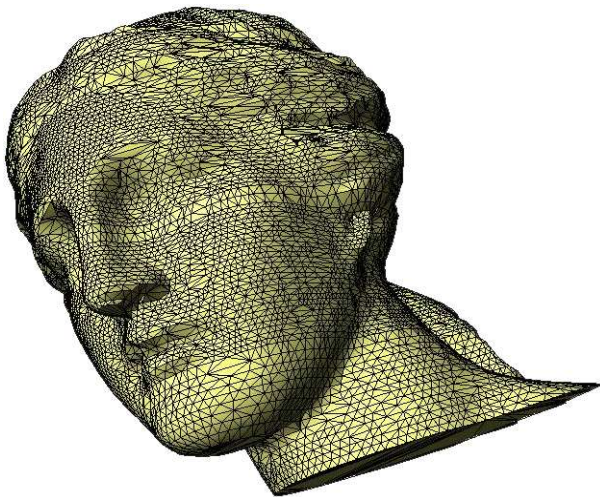
(c) Integrated Aphrodite without Border. Wireframe.

(d) Integrated Aphrodite without Border. Smooth Render.

Figure 9.   Results of Range Image Integration. Test data courtesy from Fraunhofer Inst. for Computer Graphics, Darmstadt, Germany.

intersecting Bezier curves sampled with stochastic noise. The discussion of such tests follows.

5.2.1   *Pre-processing to Transform into XY Plane.* As before, the point sample of $C(u)$ renders a quasi-planar point set. According to the discussion, an isotropic scaling was applied to the point set, because PCA is sensitive to dimensional issues. PCA was then applied to estimate the best plane Π fit to the point set, and a modified Householder transformation was used to project all points onto Π. In addition, a rigid transformation is used to bring the (now perfectly) planar point set to the XY plane, following the process described in section 4.1. Figure 10(a) shows the initial point set, along with a coordinate frame attached to the plane Π.

5.2.2   *Delaunay-based Medial Axis Processing.* The Delaunay Triangulation of the point set projected onto Π and then transformed to XY is illustrated in Figure 10(b). In the construction of the tape shaped polygon T, Delaunay Triangles included in PCA balls are accepted (Figure 10(c)). The triangles not entirely included in PCA balls may still be accepted based on the Edge Length or Area criteria (see Figure 10(d)). Notice that T is a connected

(a) Point Sample of Planar Double-8 $C(u)$ Manifold.

(b) Delaunay Triangulation of Planar Double-8 Point Sample.

(c) Filtering of Delaunay Triangulation with PCA Balls.

(d) Selected Triangles by Area and Length Criteria.

(e) Tape Polygon and its Delaunay Triangulation.
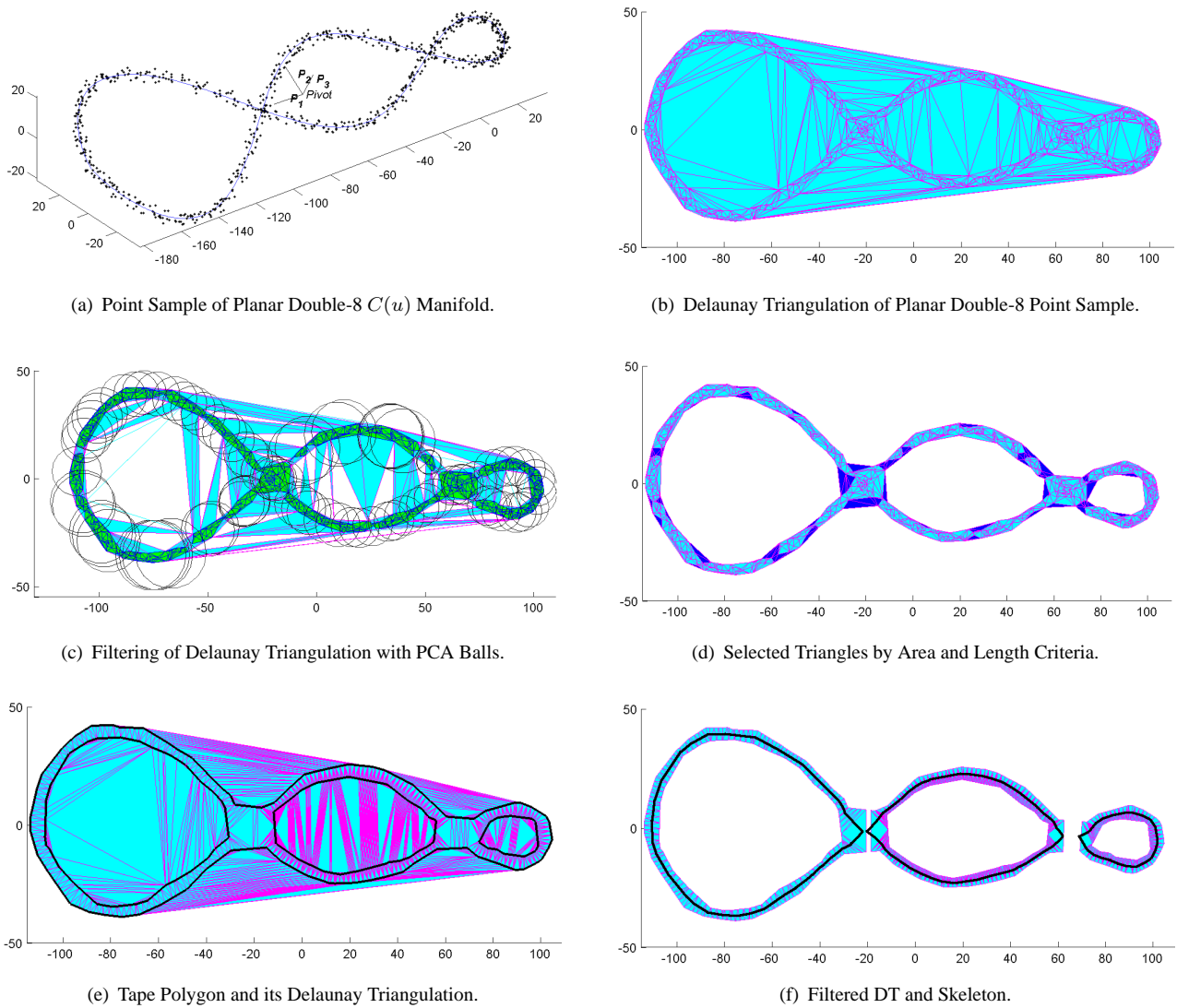
(f) Filtered DT and Skeleton.

Figure 10. Process of P.L. Approximation of Double-8 self-intersecting $C(u)$ by Combined PCA and Voronoi-Delaunay Methods.

2-dimensional region with boundary $\partial T = L_0 \cup L_1 \cup ... \cup L_m$ in Figure 4. After the region T has been synthesized by consolidating Delaunay triangles chosen according to the above criteria the border ∂T must be determined. This step is a standard procedure in Boundary Representation construction and is conducted according to the rules in section 4.4.1. The next goal is to identify the Medial Axis (MA) of T. An exact calculation is out of question because MA produces curved portions. However, if a resample RT of T is fine enough, its medial axis may be approximated as the sequence of Voronoi Edges of RT completely included in T. The *border* ∂T is resampled (see Figure 10(e)) and a new Delaunay Triangulation is calculated. The Delaunay Triangulation $DT(RT)$ of RT is purged to keep only those Delaunay Triangles internal to T. In this form, again, T is re-triangulated, but this time with triangles whose circumscribed centre lie inside T. The loci of such centres is $SK(T)$, the skeleton approximation for the medial axis $MA(T)$ of T (see Figure 10(f)). As can be seen in Figure 10(f), it is possible that the re-triangulation of T breaks this region into separate ones. This result is expected, since it indicates the presence of self-intersections in the original set. The algorithm corrects them by splitting the tape polygon T into annular sub-parts T_i. Care must still be exercised, as $SK(T)$ may be outside of a T_i region, as shown in Figure 10(f). This situation, however, is not harmful since the skeletons SK_i do not intersect each other, and therefore serve as PL approximations of the original $C_i(u)$ curves.

Figure 11 shows comparative results for a self-intersecting curve $C(u)$ (double "8") obtained using PCA (Figure 11(a)) and Voronoi-Delaunay (Figure 11(b) and Figure 11(c)) methods. Figure 11(a) shows that PCA alone processes the total point set but is not able to solve the self-intersection issue. The Voronoi-Delaunay result in
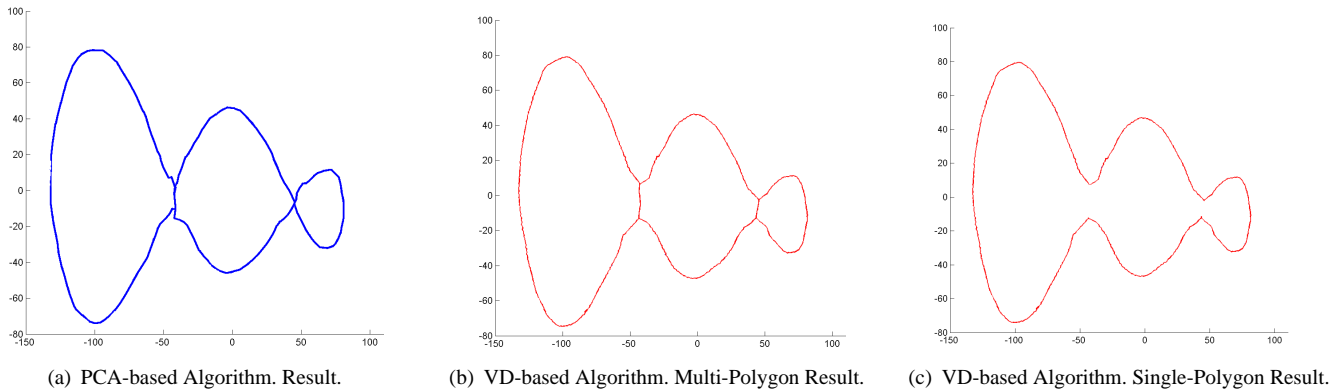
(a) PCA-based Algorithm. Result.   (b) VD-based Algorithm. Multi-Polygon Result.   (c) VD-based Algorithm. Single-Polygon Result.

Figure 11. Final Results. PL Approximations of Double-8 self-intersecting $C(u)$ by PCA and Voronoi-Delaunay Methods.

Figure11(b) solves the self-intersection by generating several tangent closed curves. The Voronoi-Delaunay result in Figure 11(c) generates a PL approximation with wasp waist.

6   Complexity Analysis

For this complexity analysis, worst-case scenarios will be considered. In the case of the Delaunay Triangulation of N points in R^2 a complexity of $O(N^2)$ is counted, instead of $O(N)$ reported in Boissonat (1998), due to the fact that no special data structure is assumed. An sketch of the complexity analysis performed is presented in the following subsections. Since only well known facts on the complexity of the Delaunay Triangulations and Graph Theory are used, the reader is invited to consult the most basic literature on such topics.

Pre-processing. Point Sample Partition. Since in both cases (self-intersecting and non self-intersecting curves) the closure operation needs to be performed, such a part is omitted in the discussion. Instead, it is assumed that a pre-process to separate all possible curve samples in the initial set is performed. Therefore, the following discussion is per curve.

6.1   *Alternative 1. Non Self-intersecting curve. PCA Analysis.*

The algorithm has a worst-case complexity of $O(N^2)$ in classifying N points in at most N balls. For each ball, the cost of PCA in a constant dimensional space (2D or 3D) is $O(N)$. Therefore, a worst-case cost of $O(N^3)$ is calculated. Figure 12 shows the execution times for the point set Aphrodite in a computer Pentium IV, Processor Clock at 3.2GHz with 2GB RAM. The curve presents an average complexity of $O(N^{1.55})$, which confirms that the expected value of complexity is much better than the worst case scenario discussed above.
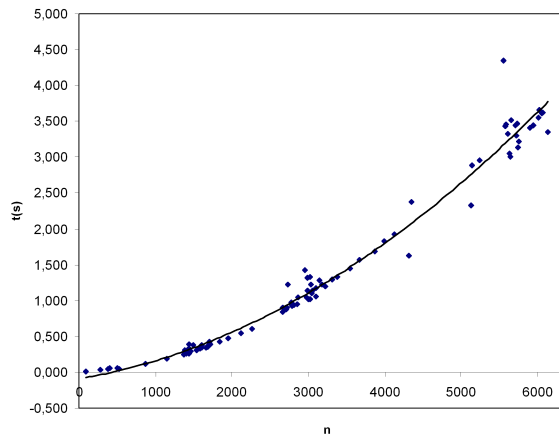

Figure 12. Execution Time of Principal Component Analysis (time vs. number of points).

6.2 *Alternative 2. Self-intersecting curve. Delaunay Triangulation and Medial Axis*

The complexity analysis for the approximation of the skeleton of the tape polygon T follows.

(i) Initial Delaunay Triangulation. First box in Figure 4. The number of triangles is $O(N)$. Cost: $O(N^2)$.

(ii) First Purge Process (using only edge length and area criteria) in a set of N triangles. Part of second box in Figure 4. Cost: $O(N)$.

(iii) Determination of $\partial T = L_0 \cup L_1 \cup .....$ from a set of N triangles. Part of second box in Figure 4. Cost: $O(N^2)$.

(iv) Resampling of each edge of ∂T in k points. Part of third box in Figure 4. Cost: $O(k.N)$.

(v) Second Delaunay Triangulation for a set of $k.N$ points, giving $O(k.N)$ as the number of triangles. Part of the third box in Figure 4. Cost: $O(k^2.N^2)$.

(vi) Second Purge Process, to see which ones of $k.N$ triangles fall inside T (T is already known from step (iii)). In the worst case, one has $O(k.N)$ as the number of vertices of the skeleton. Part of third box in Figure 4. Cost: $O(N^2)$.

(vii) Construction of the Skeleton Graph with $O(k.N)$ vertices. The initial point sample for the self-intersection curve respects the Nyquist criterion (the level of stochastic noise is smaller than half of the minimal geometric detail to be sampled). Fourth box in Figure 4. Cost: $O(k^3.N^3)$.

In conclusion, the whole process costs $O(k^3.N^3)$ if the initial curve is self-intersecting, with the construction of the final graph being the most expensive part.

7   Conclusions and Future Work

Two methods have been presented for obtaining the PL approximation of a collection of planar regular curves $C(u)$ stochastically sampled. The Principal Component Analysis -PCA- method is useful for cases when the point set corresponds to a sample of non self-intersecting curves. This method returned correctly reconstructed PL 1-manifolds for non-trivial point sets (open, unorganised, noisy, non-uniform, non-smooth, near self-intersecting).

A new application of the PCA method for surface reconstruction from Range Imaging is also discussed, and results for a real model are presented. The integration method correctly merged together a set of meshes obtained from several individual range images, into a single mesh. This approach of merging individual meshes from range pictures overcomes some of the limitations present in common usage methods based on the direct meshing from the integrated point cloud from the range pictures. The direct methods do not render a manifold topology even when the model sampled is a manifold. Our method always renders a manifold provided that it works on a Nyquist sample.

The second method (Delaunay-based Medial Axis) can be used when self-intersecting curves have been sampled, and therefore when the PCA algorithm is not applicable. This new method synthesizes the $SK(T)$ skeleton of the tape-shaped 2D region covering the point set S. This skeleton is a 1-manifold for Nyquist samples of the curve. The existing literature has not considered the reconstruction from samples of self-intersecting (or non-manifold) curves.

Future Work. When the point sample of a self-intersecting curve has low quality, building a graph, which is the PL approximation of the curve, out of the medial axis of the tape polygon T covering the curve needs improvement. In this case the graph representing the principal shape presents "hair", (i.e. high frequency artifacts), that need to be eliminated.

REFERENCES

Adamson A., and Alexa M., Anisotropic Point Set Surfaces. In *Afrigaph '06: Proceedings of the 4th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa,* 2006, p. 7–13.

Althaus, E., Mehlhorn, K., Näher, S. and Schirra, S., Experiments on Curve Reconstruction. In *ALENEX 2000*, p. 103–114, 2000.

Althaus, E., and Mehlhorn, K., Polynomial time TSP-based curve reconstruction. In *Symposium on Discrete Algorithms (SODA)*, p. 686–695, 2000.

Amenta, N., Bern, M. and Eppstein, D., The Crust and the Beta-Skeleton: Combinatorial Curve Reconstruction. *Graphical Models and Image Processing: GMIP,* 1998, **60**(2), 125–153.

Boissonnat, J.D., Shape reconstruction from planar cross-sections. *Computer Vision, Graphics and Image Processing,* 1988, **44**, 1–29.

Boissonnat, J.D., and Yvinec, M., *Algorithmic Geometry*, 1998 (UK: Cambridge University Press).

Cheng, S.W., Funke, S., Golin, M., Kumar, P., Poon, S.H. and Ramos, E., Curve reconstruction from noisy samples. *Computational Geometry,* 2005, **31**, 63–100.

Curless, B., and Levoy, M., A Volumetric Method for Building Complex Models from Range Images. *Computer Graphics,* 1996, **30**, 303–312.

Dey, T.K., and Kumar, P., A simple provable algorithm for curve reconstruction. In *10th. Annual ACM-SIAM Symposium on Discrete Algorithms*, 1999.

Edelsbrunner, H., and Mücke, E.P., Three-dimensional alpha shapes. *ACM Transaction on Graphics,* 1994, **13**, 43–72.

Edelsbrunner, H., Shape reconstruction with the Delaunay complex. In *LATIN'98: Theoretical Informatics, volume 1380 of Lecture Notes in Computer Science*, p. 119–132, 1998.

Fang, L., and Gossard, D.C., Fitting 3D curves to unorganized data points using deformable curves. In *Visual Computing, Proceedings of CG International*, p. 535–543, Berlin, 1992.

Fomenko, A., and Kunii, T., *Topological Modeling for Visualization*, 1997 (Tokio: Springer Verlag).

Fortune S., Voronoi Diagrams and Delaunay Triangulations. In *Du DZ, Hwang F, editors. Computing in Euclidean Geometry, Lecture Notes Series on Computing,* World Scientific, 1992.

Geiger, B., Three dimensional modeling of human organs and its application to diagnosis and surgical planning. 1993, Research Report 2105, INRIA, Sophia-Antipolis, Valbonne, France.

Guibas, L., and Stolfi, J., Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Transactions on Graphics,* 1985, **2**(4), 74–123.

Hastie, T., and Stuetzle, W., Principal curves. *Journal of the American Statistical Association,* 1989, **84**, 502–516.

Hoppe, H., DeRose, T., Duchamp, T., McDonald J. and Stuetzle, W., Surface Reconstruction from Unorganized points. In *ACM SIGGRAPH. 19th Annual Conference on Computer Graphics and Interactive Techniques*, p. 71–78, Chicago, 1992.

Kegl, B., Principal Curves: Learning, Design, and Applications. PhD thesis, Concordia University, Montreal, Canada, 1999.

Kegl, B., and Krzyzak, A., Piecewise Linear Skeletonization Using Principal Curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2002, **24**(1), 59–74.

Lee, I.K., Curve reconstruction from unorganized points. *Computer Aided Geometric Design,* 2000, **17**(2), 161–177.

Mantyla, M., *An Introduction to Solid Modeling*, 1988 (Maryland: Computer Science Press).

Morse, M., *The calculus of variations in the large*, 1934 (New York: American Mathematical Society).

Ogniewicz, R., Skeleton-Space: a Multiscale Shape Description Combining Region and Boundary Information. In *IEEE Conference on Computer Vision and Pattern Recognition*, p. 746–751, Seattle, 1994.

Ohtake, Y., Belyaev, A. and Seidel, H., Skeleton-Space: n Integrating Approach to Meshing Scattered Point Data. In *SPM '05: Proceedings of the 2005 ACM Symposium on Solid and Physical modeling*, p. 61–69, Cambridge, 2005.

Raindrop Geomagic®Inc. *Studio 8*. http://www.geomagic.com, ©2006.

Ruiz, O.E., Cadavid, C.A., Granados, M.A., Peña, S. and Vásquez, E. 2D shape similarity as a complement for Voronoi–Delone methods in shape reconstruction. *Elsevier J. on Computers and Graphics,* 2005, **29**(1), 81–94.

Soucy, M., and Laurendeau, D., A General Surface Approach to the Integration of a Set of Range Views. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 1995, **17**(4), 344–358.

Taubin, G., and Ronfard, R., Implicit simplicial models for adaptive curve reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 1996, **18**(3), 321–325.

Turk, G., and Levoy, M., Zippered Polygon Meshes from Range Images. In *SIGGRAPH'94: Computer Graphics Proceedings, Annual Conference Series*, p. 311–318, Orlando, Florida, 24–29 July, 1994.

Varady, T., Martin, R. and Cox, J., Reverse Engineering of Geometric Models. An introduction. *Computer Aided Design,* 1997, **29**(4), 255–268.

Verbeek, J., Vlassis, N. and Kröse, B., A Soft k-Segments Algorithm for Principal Curves. In *Dorffner G, Bischof*

H, Hornik K, editors. ICANN 2001, LNCS 2130, 2001, p. 450–456.

Wang, W., Pottmann, H. and Liu, Y., Fitting B-spline curves to point clouds by curvature-based squared distance minimization. *ACM Transactions on Graphics,* 2006, **25**(2), 214–238.

Zhou, H., Liu, Y. and Li, L., Incremental Mesh-based Integration of Registered Range Images: Robust to Registration Error and Scanning Noise. In *Seventh Asian Conference on Computer Vision, Hyderabad, India*, 2006, p. 958–968.