Skeleton Extraction from Point Clouds based on Discrete Curvature

Hyoungseok B. Kim† and Hosook Kim††,

Department of Multimedia Engineering, Dong-Eui University, Busan, Rep. of Korea
Division of Mathematics and Computer Science, KSA of KAIST, Busan, Rep. of Korea

Summary
We present an algorithm to extract the skeleton from unorganized point clouds by exploiting a discrete curvature. In general, the result of skeleton extraction depends on the ordering of sample points and the geometric properties of consecutive points in the order. In this paper, we utilize the discrete curvature defined by three consecutive sample points to extract an order from points. Our method is more intuitive than the previous methods and can extract the skeleton from point clouds which resembles the global shape of the clouds.

Key words:
Curve reconstruction, discrete curvature.

1. Introduction

The problems of extracting a skeleton from sample points appear in many scientific and engineering applications. Because of its practical importance, many algorithms have been proposed over the last two decades in the fields such as the reverse engineering of geometric models and medical imaging. Especially, curve reconstruction plays an important role in the shape reconstruction problems [1, 2].

Curve reconstruction is the problem of computing a piecewise linear approximation to a curve from a set of sample points. Many approaches have been suggested for the reconstruction of curves from sample points. Edelsbrunner et al. [3] defined the alpha-shapes of point sets as the underlying space of a simplicial complex. Attali [4] proposed the r-regular shape method, where the r-regular shapes are characterized by requiring that any circle passing the points on the boundary has radius greater than r. These methods deal with only uniform sample points. For the non-uniform sample points, Amenta et al. [5] proposed the first algorithm to reconstruct a curve from non-uniform sample points with guarantee. This algorithm uses the Voronoi diagram and the Delaunay triangulation of the sample points. Dey et al. [6] proposed the nearest neighbor approach based on the properties of Voronoi diagrams. These algorithms work only under the assumption that the sample points are dense and do not work for non-simple curve reconstruction because they reconstruct curves without the consideration of curve's orientation. There has been little research for curve reconstruction from an unorganized point set. Fang et al. [7] used a method based on spring energy minimization to approximate an unorganized point set with a curve, which needs a good initial guess of the solution. Dedieu et al. [8] presented an algorithm for ordering unorganized points assuming that all points are on the reconstructed curves. This is not appropriate for a point cloud. Taubin et al. [9] reconstructed a planar curve from unorganized points using an implicit simplicial curve, which is defined by a planar triangular mesh and the values at the vertices of the mesh. These methods are not appropriate for curves with self-intersections. Most of all curve reconstruction algorithms are based on the Euclidean distance to compute the proximity and the adjacency. Recently, Kim et al. [10] proposed a method based on Brownian motion, which can reconstruct curves with self-intersection.

In this paper, we propose a curve reconstruction algorithm based on discrete curvature, which extracts the representative points from point clouds so that the shape of the reconstructed curve resembles that of the set of point clouds. This paper is organized as follows. The concept of discrete curvatures is introduced in Section 2. The algorithm is explained in Section 3. The algorithm utilizes the principal component analysis in order to choose the initial direction of reconstructing a curve, and the greedy approach in each step in order to select the next point of minimal curvature among the candidates. Section 4 shows the experimental results. In Section 5, we conclude this paper with some remark on future researches.

2. Discrete Curvature

In this section, we introduce a discrete curvature estimation proposed by Kim et al. [11] which is based on the parabolic interpolation. In general, the local shape of a polygon at a vertex is determined by the geometric relationship between the vertex and its adjacent vertices. We have known that the concept of curvature is derived

Manuscript received November 5, 2011
Manuscript revised November 20, 2011

from a curve and it is a quantity to measure the local bending of curves. Therefore, the best method to resemble the local shape, following the original definition of a curvature is to use the quadratic curve interpolating the three consecutive vertices.

We adopt a quadratic Bezier curve as an interpolating curve. Let P, Q, R be three consecutive vertices. The general form of the quadratic Bezier curve satisfying $B(1/2) = Q$ is as follows:

$$B(t) = PB_0^2(t) + CB_1^2(t) + RB_2^2(t),$$

where $B_i^n(t) = \frac{n!}{(n-i)!i!}(1-t)^{n-i}t^i$ are the Bernstein polynomials of degree n and $C = (4Q - P - R)/2$. Figure 1 shows the three control points of the Bezier curve.

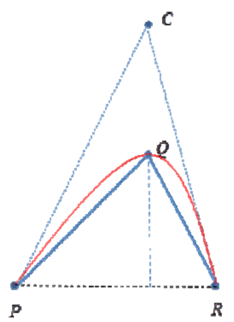


Fig. 1 Discrete curvature of three consecutive points P, Q, R.

The curvature of $B(t)$ at $t = 1/2$ is

$$\mathrm{K}_P(1/2) = \frac{\left\| B''(1/2) \times B'(1/2) \right\|}{\left\| B'(1/2) \right\|^3}$$

Hence, we can define a new Parabola-based discrete curvature of the given vertex Q as follows:

$$curv_P(Q) = \frac{\left\| 4(P - 2Q + R) \times (R - P) \right\|}{\left\| R - P \right\|^3}. \qquad (2)$$

First of all, we find out the geometric properties of the P-discrete curvature formula. Let $V = (R - P)/2$ and $G = P - 2Q + R$. The P-discrete curvature formula is

$$curv_P(Q) = \frac{\left\| 4G \times 2V \right\|}{\left\| 2V \right\|^3} = \frac{\left\| G \right\| \left\| V \right\| \sin\theta}{\left\| V \right\|^3} \qquad (3)$$

where θ is the in-between angle of the vectors G and V.

3. Practical Algorithm

In this section, we will describe a practical algorithm for ordering points in an unorganized point cloud. Assume that there are arbitrarily scattered \$n\$ points, called *sample points*, in R^3. Denote the set of all sample points by P. The point ordering problem aims to choose a subset sP of P, where sP is a well-ordered set. In the above theoretical section, we have assumed that the first point p_0 and the second point p_1 are known. This easily enabled us to initialize the algorithm at the first point with an initial direction p_0p_1. In general, the result of the point ordering problem is dependent on the initial direction. But, in practice, p_0 and p_1 are not known. Moreover, the well-ordered subset sP may have several connected components, and may be open or closed according to the distribution of the data points. So we need to analyze the distribution of the sample points in order to find out the property of the subset sP. In general, the sample points of the unorganized point cloud for curve reconstruction may cluster in several groups and the points of each group are densely located near meaningful trajectories. So, the shape of each cluster looks like the union of bands. The objective of our algorithm is to remove outliers in each cluster and find out a well-ordered subset that plays a role of skeletons of clusters. Our algorithm consists of three major steps: data clustering, determination of the initial direction, and local point ordering.

Data Clustering

Most partitioning methods cluster objects based on the Euclidean distance between objects. Such methods can find only spherical-shaped clusters and encounter difficulty at discovering clusters of arbitrary shapes. However, the shape of the input point set P is unknown we cannot apply such clustering methods. In this paper, we adopt the *DBSCAN* clustering algorithm based on the density, which is useful to filter out noises (outliers) and discover clusters of arbitrary shape. The general idea is to continue growing the given cluster as long as the density in the neighborhood exceeds some threshold; that is, for each data point within a given cluster, the neighborhood of a given radius cr, called *clustering radius* has to contain at least a minimum number of points. By exploiting *DBSCAN* algorithm, we may obtain nc clusters $CL_1, CL_2, CL_3, \cdots, CL_{nc}$. For each cluster CL_j, we choose a point which may well represent the character of the shape of the cluster as the initial point, denoted by cp_j.

Determination of Initial Direction

Now, we will try to determine the initial direction of the natural distance at the initial point cp_j for each cluster CL_j. For the sake of convenience, we set the initial point as p_0. In order to analyze the distribution of points near p_0, we compute the neighborhood $N(p_0,r)$ of p_0 with a given radius r, called *ordering radius*;

$$N(p_0,r) = \{p \in CL_j \mid d(p_0,p) < r\},$$

where $d(p,q)$ is the Euclidean distance between two points p and q. And then we apply the PCA (*Principal Components Analysis*) to the neighborhood because the principal component analysis of a set of points gives us the mean, an orthogonal frame, and the standard deviation of the neighbors. Let $q_1, q_2 \cdots, q_m$ be the points in the neighborhood $N(p_0,r)$. We compute the mean m of the neighbors such that $m = (q_1 + q_2 + \cdots + q_s)/s$ and then we construct the covariance matrix $\mathbf{C}$ by

$$\mathbf{C} = \frac{1}{s}\sum_{j=1}^{s}(q_j - m)(q_j - m)^T = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix},$$

where $m = (m_x, m_y), q_j = (q_{x_j}, q_{y_j}),$

$$C_{11} = \frac{1}{s}\sum_{j=1}^{s}(q_{x_j} - m_x)^2,$$

$$C_{12} = C_{21} = \frac{1}{s}\sum_{j=1}^{s}(q_{x_j} - m_x)(q_{y_j} - m_y)$$

$$C_{22} = \frac{1}{s}\sum_{j=1}^{s}(q_{y_j} - m_y)^2$$

We compute the eigenvector of the covariance matrix which corresponds to the maximal eigenvalue so that it plays a role of the major principal axis in the neighborhood. Let ID be the unit vector on the principal axis. Then the neighborhood $N(p_0,r)$ may be subdivided into two half discs:

$$N(p_0,r) = HD_-(p_0,r) \cup HD_+(p_0,r),$$

where

$$HD_-(p_0,r) = \{p \in N(p_0,r) \mid p_0p \cdot ID < 0, p \neq p_0\},$$
$$HD_+(p_0,r) = \{p \in N(p_0,r) \mid p_0p \cdot ID > 0, p \neq p_0\}$$

If $HD_- = \phi$ or $HD_+ = \phi$, then the point p_0 is one of the two end points of sP. Otherwise the point is a middle point of sP so that we should apply the following one-way ordering algorithm to both HD_+ and HD_- with the direction vectors ID and $-ID$, respectively. Let sP_+ and sP_- be the solutions of the one-way ordering problem with the initial directions ID and $-ID$, respectively. Then the solution sP of the original point ordering problem may be obtained by merging the subsets sP_+ and sP_-, i.e., $sP = sP_+ \cup sP_-$.

Local Point Ordering

Now, we are ready to explain the one-way point ordering algorithm with the first point p_0 and the initial direction ID. First of all, we have to select the candidates of the next point p_1 in the neighborhood $N(p_0,r)$. It is well known that the candidate set C_1 is the same as the half disc $HD_+(p_0,r)$. So, without loss of generality, we may put $C_{i+1} = HD_+(p_i,r)$. Next, we select the second point p_1 in C_1 satisfying the following condition:

$$curv_{p_0}(p_1) = \min\{curv_{p_0}(p) \mid p \in HD_+(p_0,r)\},$$

where the function $curv_{p_0}(p)$ is derived from the initial direction vector ID. The point p_1 is added to the well-ordered set so that $sP_+ = \{p_0, p_1\}$. In order to find out the third point p_2, we compute the candidate set C_2 that is the half disc $HD_+(p_1,r)$ with the direction vector $D = [p_0p_1]$. The third point p_2 should satisfy the following condition:

$$curv_{p_1}(p_2) = \min\{curv_{p_1}(p) \mid p \in HD_+(p_1,r)\},$$

where the function $curv_{p_1}(p)$ is derived from the direction vector $D = [p_0p_1]$. Then $sP_+ = \{p_0, p_1, p_2\}$. We apply the above process until the candidate set C_{i+1} is empty or the intersection of C_{i+1} and $HD_-(p_0)$ is not empty. When the algorithm meets the former condition, it generates one of the two end points for an open curve, so it has to be applied to the other region of the cluster with the direction $-ID$. If the latter is satisfied, then the new obtained point p_{n_1} is near the first point p_0. We can consider the well-ordered set is a closed piecewise curve so that there is no need to continue this one-way point ordering algorithm in the cluster. The solution of the one-way point ordering problem is

$$sP_+ = \{p_0, p_1, p_2, \cdots, p_{n_1}\}.$$

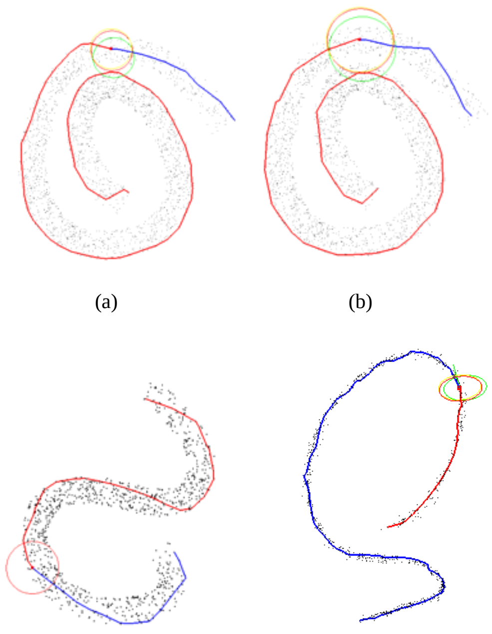
By similar process, we may get

$$sP_- = \{p_{-n_2}, p_{-n_2+1}, \cdots, p_{-1}\}$$

as the solution of the backward one-way ordering algorithm. Now, we may obtain the first connected well-ordered subset $sP = sP_+ \cup sP_-$. Therefore, the global point ordering problem can be solved by applying the above process to all of clusters in sP.

4. Experimental Results

In this section, we tested the performance of our algorithm for several reconstruction problems where each problem is focused on the following facts: (1) the effect of clustering radius in point clouds, (2) the effect of ordering radius, and (3) the ability of reconstructing a 3D curve from 3D points sampled on the quadratic surfaces.

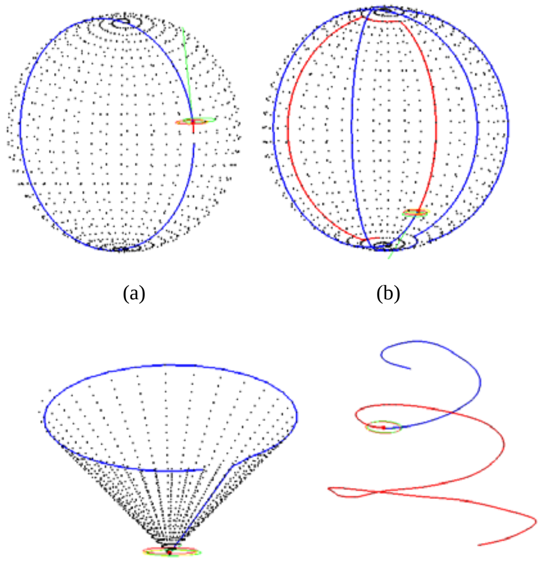


(a) (b)

(c) (d)

Fig. 2 Results for 3D point clouds

There are two different results of extracting the skeleton from point clouds according to the size of the clustering radius as shown in Figure 2. The radii of circles of red color and green color represent the sizes of ordering radius and clustering radius, respectively. As shown in Figure 2 (a) and (b), the smaller the size of clustering radius is, the smoother the extracted skeleton is. The blue colored curve represents the subset sP_+, and the red colored curve does sP_-.



(a) (b)

(c) (d)

Fig. 3 3D curve reconstruction of sample points

Figure 3 presents four piecewise linear curves obtained by our algorithm with the clustering radius $cr = 0.5$. Figure 3(a) and (b) shows the effect of the size of the ordering radius. The size of ordering radius of Figure 3 (a) is greater than that of Figure 3 (b). The points near the North pole of Figure 3 (a) was already used in local point ordering so that the points cannot be candidates in the next ordering and thus the shape of curve reconstruction likes a circle. Whereas the points near the North pole of Figure 3 (b) can be candidates for the next ordered points in local point ordering, so that the reconstructed curve can pass through near North pole in several times.

5. Conclusion

We proposed a new method for constructing curves from unorganized point clouds with noise. In general, the result of curve reconstruction depends on how to select and order the representative points to resemble the shape of the clouds. In this paper, we exploit discrete curvature to reflect orientation to the ordering of sample points, so that our algorithm is able to reconstruct not only simple curves but also non-simple curves. Moreover, for unorganized point clouds, this method can efficiently extracts the skeletons of the clouds by cutting out the outliers, even though the result by our method is sensitive to the initial point and the initial direction. Our algorithm consists of three steps: point clustering, initialization of point and direction, and local point ordering. The core of our algorithm is the third step, local point ordering, which adopts the discrete curvature as a measure of what is the best next point.

References

[1] Hoppe, H. and DeRose, T. and Duchamp, T. and McDonald, J. and Stuetzle, W., "Surface Reconstruction from unorganized points", SIGGRAPH 92, pp 71-78, 1992.

[2] Boissonnat, J.D. and Geiger, B., "Three-dimensional reconstruction of complex shapes based on the Delaunay triangulation", Proc. of Biomedical Image Process. Biomed. Visualization, pp 964-975, 1993.

[3] Edelsbrunner, H. and Kirkpatrick, D.G. and Seidel, R., "On the shape of a set of points in the plane", IEEE Trans. Information Theory, pp 551-559, 1983 Attali, D., "r-regular shape reconstruction from unorganized points", Proc. of 13th Ann. Sympos. Comput. Geom., p 248-253, 1997.

[4] Attali, D., "r-regular shape reconstruction from unorganized points", Proc. of 13th Ann. Sympos. Comput. Geom., pp 248-253, 1997.

[5] Amenta, N. and Bern, M. and Eppstein, D., "The crust and the beta-skeleton: combinatorial curve reconstruction", Graphical Models, Vol. 60, pp 125-135, 1998.

[6] Dey, T.K. and Wenger, R., "Reconstructing curves with sharp corners", Comput. Geom. - Theory Appl., Vol. 19, pp 89-99, 2001

[7] Fang, L. and Gossard, D.C., "Fitting 3D curves to unorganized data points using deformable curves", Proc. of CG International 92, pp 535-543, 1992.

[8] Dedieu, J.P. and Favardin, C.H., "Algorithms for ordering unorganized points along parameterized curves", Numerical Algorithms, Vol. 6, pp 169-200, 1994.

[9] Taubin, G. and Ronfard, R., "Implicit simplicial models for adaptive curve reconstruction", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 18, pp 321-325, 1996.

[10] Kim, Philsu and Kim, Hyoungseok "Point Ordering with Natural Distance Based on Brownian Motion", Journal of Mathematical Problems in Engineering, Vol 20, Article ID 450460, 2010

[11] Kim, Hyoungseok and Kim, Hosook, "New Computation of Normal Vector and Curvature", WSEAS Transactions on Computers, Vol. 8, No. 10, pp 1661-1670, 2009

Hyoungseok B. Kim received the M.S. and Ph.D. degrees in department of Mathematics from Korea Advanced Institute of Science and Technology in 1992 and 1998, respectively. During 1998-1999, he worked at Electronics and Telecommunications Research Institute (ETRI) as a post-doctorial researcher. He is a professor in department of Multimedia Engineering at Dong-Eui University from 1999. His interesting research fields are computer graphics, computer game engine, and applied mathematics.

Hosook Kim received the M.S. and Ph.D. degrees in department of Computer Engineering from Ewha Womans University in 1999 and 2005, respectively. She was an associate professor in division of Advance Computer Information at Dong-Eui Institute of Technology from 2001 to 2009. She is a teacher in the division of mathematics and computer science at Korea Science Academy of KAIST from 2010. Her interesting research fields are database, data mining, GIS, and mobile technology.