

Grouping and Parameterizing Irregularly Spaced Points for Curve Fitting

A. ARDESHIR GOSHTASBY

Wright State University

Given a large set of irregularly spaced points in the plane, an algorithm for partitioning the points into subsets and fitting a parametric curve to each subset is described. The points could be measurements from a physical phenomenon, and the objective in this process could be to find patterns among the points and describe the phenomenon analytically. The points could be measurements from a geometric model, and the objective could be to reconstruct the model by a combination of parametric curves. The algorithm proposed here can be used in various applications, especially where given points are dense and noisy. Examples demonstrating the behavior of the algorithm under noise and density of the points are presented and discussed.

Categories and Subject Descriptors: I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling—*Curve, surface, solid, and object representations*

General Terms: Algorithms

Additional Key Words and Phrases: Irregularly spaced points, node estimation, noisy point set, parametric curve

1. INTRODUCTION

In many science and engineering problems there often is a need to fit a curve or curves to an irregularly spaced set of points. Curve fitting has been studied considerably in approximation theory and geometric modeling and many books on the subject [Beach 1991; Farin 1988; Faux and Pratt 1979; Hagen 1992; Lancaster and Salkauskas 1986] have been published. Existing techniques typically find a single curve segment that approximates or interpolates a set of points. Many techniques assume that the points are ordered and fit a curve to them by minimizing an error criterion [Cohen and O'Dell 1989; Foley and Nielson 1989; Fritsch and Carlson 1980; Hartley and Judd 1980; Hölzle 1983; Hoschek 1987; Kosters 1991; Marin 1984; Mullineux 1982; Sarkar and Menq 1991]. If the points are ordered,

Author's address: Computer Science and Engineering Department, Wright State University, Dayton, OH 45435.

Permission to make digital/hard copy of part or all of this work for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee.

© 2000 ACM 0730-0301/00/0700-0185 \$5.00

piecewise polynomial curves can be fitted to them also [Ichida and Kiyono 1977; Pieggl 1987]. Difficulties arise when the points are not ordered.

We assume that a large set of irregularly spaced points in the plane, $\{\mathbf{p}_i = (x_i, y_i) : i = 1, \dots, N\}$, is given and it is required to fit one or more parametric curves to the points, with the number of the curves to depend on the organization of the points and the resolution of the representation. When fitting a parametric curve to an irregularly spaced set of points, the main problem is to find the nodes of the curve. The nodes of a parametric curve determine the adjacency relation between the points and order them. The curve then approximates the points in the order specified. Methods to order sparse point sets have been available for some time [Farouki 1997; Grossman 1970; Hoschek 1988a; Jüttler 1997; Lee 1989; Speer et al. 1998]. Methods to order dense and noisy point sets have also recently surfaced.

To order a set of irregularly spaced points, Levin [1998] used moving least squares to reorganize the points into a thinner set. For each point in the set, a curve that fits to that point and points in its neighborhood is computed using a weighted regression method. Then the point is moved to the approximating curve. If the points are widely spread, the curve points determined in this manner may still fall apart, forming an irregular but thinner point set. To obtain a sufficiently thin point set where ordering of the points is straightforward, several iterations of the moving least squares may be necessary. The size of the neighborhood chosen in the regression method influences the outcome of the thinning process. Lee [2000] studied the effect of neighborhood size on the thinning process and suggested optimal neighborhood sizes for the points in a set to produce sufficiently thin point arrangements to facilitate ordering of the points.

Methods to fit a parametric curve to irregularly spaced points without ordering the points have been proposed also. Pottmann and Randrup [1998] mapped a point set into a digital image by quantizing the coordinates of the points with an appropriate quantization step. The binary region obtained in the digital image was then thinned to obtain the medial axis of the region, and a curve was fitted to pixels in the medial axis. Hastie and Stuetzle [1989] called the trajectory formed by averaging local points in the direction normal to the trajectory the principal curve of the points. To determine the principal curve of a set of points, they started with an approximating curve and iteratively revised the curve by projecting the points to it until the curve fully converged to the data. To make the principal curve better represent the local geometry of a set of points, Banfield and Raftery [1992] removed the outliers. This also improved the bias and variance of the curve.

Existing methods fit a single curve segment to a given set of points. Sometimes it is not appropriate to fit a single curve segment to a very large and complex point set. The point set may contain many geometric structures, each requiring a curve segment to represent it. To fit curves to an irregularly spaced set of points, (1) the set should be partitioned into

subsets, (2) the points in each subset should be ordered, and (3) a curve should be fitted to the points in each subset. This article provides solutions to the first two problems, that is, partitioning a point set into subsets and ordering the points in each subset. Once the points in each subset are ordered, existing techniques can be used to find the curves. A curve that approximates a set of dense and noisy points should trace the spine of the points, and if the points are noisy, the process should perform some kind of an averaging to reduce the noise.

In the model proposed in this article, a radial field is centered at each point such that the strength of the field monotonically decreases as one moves away from the point. The sum of the fields has the averaging effect and reduces the effect of noise, and local maxima of the sum of the fields has the effect of tracing the spine of the points. The local maxima of the sum of the fields (the ridges of the obtained field surface) are used as approximations to the curves to be determined. Based on the organization of the points, many ridges may be obtained, each suggesting a curve. The ridges are used to partition the points into subsets and to fit a curve to each subset.

2. DEFINITIONS

The following definitions are used to describe the proposed approach.

Given points refers to a set of dense and noisy points in the plane, $\{(x_i, y_i) : i = 1, \dots, N\}$, provided in no particular order.

Inverse distance between points (x, y) and (x_i, y_i) is defined by

$$[(x - x_i)^2 + (y - y_i)^2]^{-\frac{1}{2}}. \quad (1)$$

A *digital image* I is a 2-D array of values representing isotropic samples from a continuous explicit function $z = g(x, y)$ at equally spaced intervals along x and y axes. Entry (i, j) in the image shows the value of function g at the j th sample point in the i th scanline. An entry of the image is referred to as a *pixel*.

Gradient direction at a pixel is the direction of maximum intensity change at the pixel. In a digital image, gradient direction can have one of eight possible values. These are the directions obtained by connecting a pixel to one of its eight neighbors (see Figure 1a).

Gradient magnitude at a pixel is the magnitude of intensity change in the gradient direction.

Minor ridges in an image refer to pixels whose values are larger than those of pixels adjacent to them and at opposite sides to each other. Pixel A is a minor-ridge pixel if pixels B and C adjacent to it and at opposing sides of A have intensities that are smaller than that at A . More

specifically, pixel (i, j) is marked as a minor-ridge pixel if one of the following is true.

$$I(i - 1, j) < I(i, j) \text{ and } I(i + 1, j) < I(i, j); \quad (2)$$

$$I(i, j - 1) < I(i, j) \text{ and } I(i, j + 1) < I(i, j); \quad (3)$$

$$I(i - 1, j - 1) < I(i, j) \text{ and } I(i + 1, j + 1) < I(i, j); \quad (4)$$

$$I(i - 1, j + 1) < I(i, j) \text{ and } I(i + 1, j - 1) < I(i, j). \quad (5)$$

The minor ridges of the image of Figure 1(b) are shown in Figure 1(c).

Major ridges are pixels in an image whose intensities are higher than those of pixels adjacent to them in the gradient direction. To find the major ridges in an image, we have to find each pixel D where not only its intensity is larger than two pixels B and C adjacent to it and at opposing sides of A , but which also has a gradient direction that is the same as the direction obtained by connecting B to C . Major ridges are a subset of minor ridges. The major ridges of the image of Figure 1(b) are shown in Figure 1(f).

A *minor-ridge segment* is a connected set of minor-ridge pixels where each pixel has at most two neighbors. For example, the minor ridges of Figure 1(d) have the minor-ridge segments shown in Figure 1(e). A through F are the minor-ridge segments.

A *branch pixel* is a minor-ridge pixel that is also the end pixel of three or more minor-ridge segments. A branch pixel, therefore, has more than two neighbors. In Figure 1(e), M and N are the branch pixels. The branch pixels provide the means to cut the minor ridges into ridge segments.

A *ridge contour* is a set of connected minor-ridge segments, each of which contains at least one major-ridge pixel. A ridge contour may be formed from many ridge segments and, therefore, it may contain branch pixels. But no pixels in a ridge contour have more than two neighbors. For instance, in Figure 1(e), minor-ridge segments B , D , and F form a ridge contour if segments B , D , and F each contain at least one major-ridge pixel.

A *contour* is a sequence of eight-connected pixels in an image where each pixel can be reached from any other pixel in the contour. An open contour has two end pixels, each with only one neighbor, and zero or more interior pixels, each with exactly two neighbors. A closed contour has no end pixels. All pixels in a closed contour have exactly two neighbors.

The *length of a contour* is computed from the sum of distances between adjacent pixels in the contour. Two adjacent pixels that lie horizontally

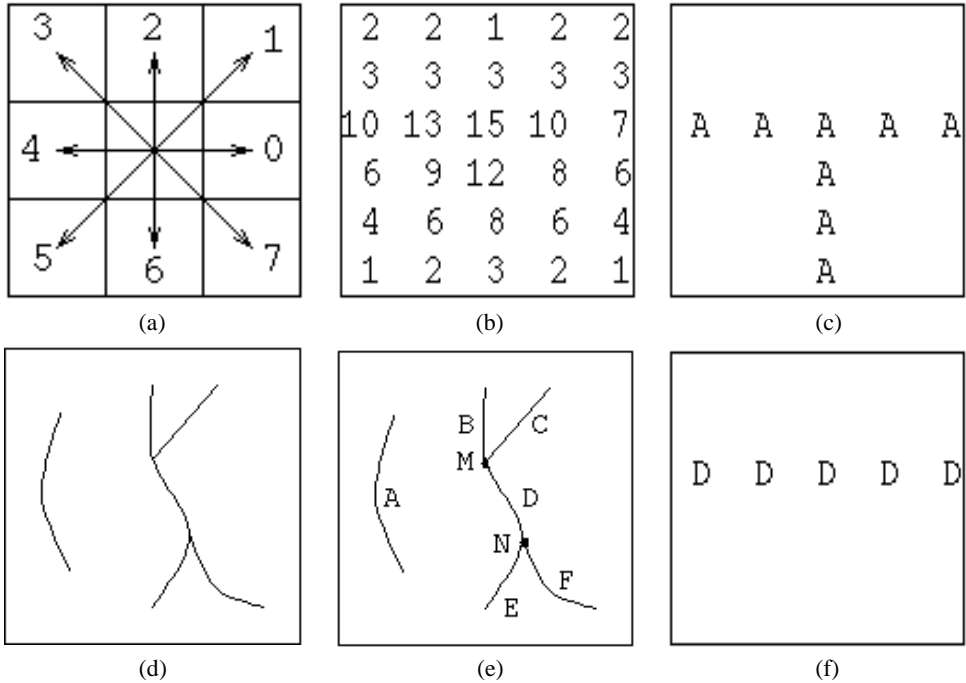


Fig. 1. (a) Possible discrete gradient directions at a pixel in a digital image; (b) a digital image; (c) minor ridges of image (b); (d) another example of minor ridges in an image; (e) A through F are the ridge segments and M and N are the branch pixels; (f) major ridges of image (b).

or vertically are considered to be 1 pixel apart. All other adjacent pixels are considered to be $\sqrt{2}$ pixels apart .

3. PROPOSED APPROACH

A desirable property of an approximating curve is to pass as closely as possible to the given points while providing a desired smoothness appearance. For a dense point set, the curve cannot pass close to all the points, so it should trace the spine of the points. In the model proposed here, initial estimations to the curves are obtained by finding points in the xy plane whose sum of inverse distances to the given points are locally maximum. Therefore, by tracing points in the xy plane that locally maximize

$$f(x, y) = \sum_{i=1}^N [(x - x_i)^2 + (y - y_i)^2 + 1]^{-\frac{1}{2}}, \tag{6}$$

we can find an approximation to the curves to be found. The addition of 1 to each term of Equation (6) is to avoid division by 0.

Function f can be interpreted as follows: suppose a radial field of strength 1 is centered at point (x_i, y_i) , $i = 1, \dots, N$, such that the

strength of the field decreases with inverse distance as one moves away from the point. Then the strength of the field at point (x, y) will be $[(x - x_i)^2 + (y - y_i)^2 + 1]^{-1/2}$, and the curves to be found can be considered points in the xy plane where the sum of the field values is locally maximum.

In order to provide an ability to control the shape or smoothness of the obtained ridges, we replace the constant 1 in each term of formula (6) with r^2 to obtain

$$[(x - x_i)^2 + (y - y_i)^2 + r^2]^{-\frac{1}{2}} \quad (7)$$

and define

$$g(x, y) = \sum_{i=1}^N [(x - x_i)^2 + (y - y_i)^2 + r^2]^{-\frac{1}{2}}. \quad (8)$$

The basis functions defined in formula (7) are the inverse multiquadric basis functions [Hardy 1990]. The parameter r of the basis functions can be varied to produce surfaces with different shapes [Kansa 1990]. Figure 2(b) shows the field surface obtained when using the data points in Figure 2(a) and the inverse multiquadric basis functions with $r = 5$.

In addition to inverse multiquadric basis functions, other radial basis functions [Buhmann 1990; Dyn 1987; Goshtasby and O'Neill 1993; Meinguet 1979; Powell 1987; Schagen 1980] can be used to define g . The selected basis functions influence the shape of the obtained field surface, the shapes of the obtained ridges, and, consequently, the shapes of the obtained curves. Later we will show how the shapes of the ridges, and therefore the shapes of the curves, change as different basis functions are used.

If the local maxima of surface g is traced in the xy plane, an approximation to the curves will be obtained. Parameter r changes the shape of the basis functions and affects the shape of the field surface. We investigate the effect of this parameter on the shapes of generated curves later in this article.

Local maxima of surface g can result in structures that contain branches and loops. The proposed model can, therefore, recover very complex patterns in dense and noisy point sets. Note also that the proposed method does not require any knowledge about the adjacency relation between the points. This method, in fact, provides the means to determine the adjacency relation between the points and order the points for curve fitting.

4. IMPLEMENTATION

Derivation of an analytic formula that describes the local maxima of surface g may not be possible. A digital approximation to the local maxima, however, can be obtained. This approximation will be in the form of digital

contours. To trace the ridges of surface g , the surface is first digitized into a digital image. The digitization process involves starting from $x = x_1$ and $y = y_1$ and incrementing x and y by some small increment δ until $x = x_2$ and $y = y_2$ are reached. For each discrete (x, y) the value for $g(x, y)$ is found from formula (8). x_1 and x_2 could be the smallest and largest x coordinates, and y_1 and y_2 could be the smallest and largest y coordinates of the given points. Parameter δ is used as the increment for both x and y because radially symmetric basis functions are used to represent surface g . This parameter determines the resolution of the obtained image. For a finer digitization, this parameter should be reduced, while for a coarser digitization this parameter should be increased. If this parameter is to be chosen automatically, it should be selected such that the majority of given points map to unique pixels in the image.

Digitizing surface g in this manner will result in a digital image whose pixel values show uniform samples from the surface. Figure 2(c) shows digitization of the field surface of Figure 2(b) into a 256×256 grid. Figure 2(d) shows this image with pixel intensities showing the heights of the surface points. The minor and major ridges of this image are shown in Figures 2(e) and 2(f), respectively. We see that the major ridge points do not fall on small and noisy branches of the minor ridges but rather fall on contours that represent the spines of the points. The minor ridges are cut at the branch points, and branches that do not contain a major ridge point are removed to obtain the ridge contours shown in Figure 2(g). These contours will be taken as approximations to the curves to be found. They will be used not only to partition the given points into subsets but to order the points in each subset and fit a curve to each subset. To summarize, the steps of the proposed grouping and curve-fitting algorithm are as follows.

- (1) From the given points, determine field surface g .
- (2) Quantize surface g into a digital image.
- (3) Find the minor ridges of the image and partition the ridges into ridge segments.
- (4) Find the major ridges of the image.
- (5) Remove the minor-ridge segments that do not contain a major ridge pixel.
- (6) Find the ridge contours from the remaining minor ridge segments.
- (7) Partition the set of points into subsets using the ridge contours and find the nodes of the points in each subset (see below).
- (8) Fit a curve to each point subset (see below).

The shapes of the ridge contours change as parameter r of the inverse multiquadrics changes. Since the ridge contours are used to partition points in a set into subsets, as the subsets change, the shapes of curves to

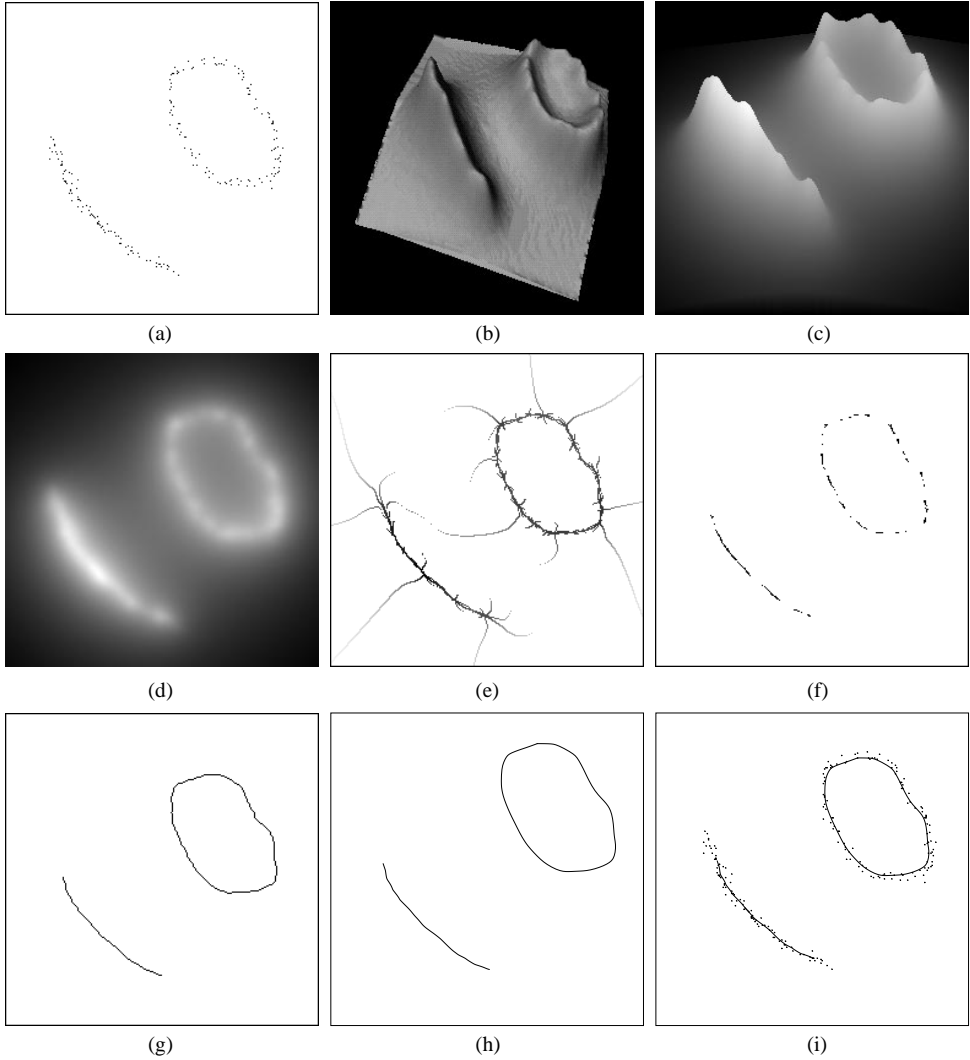


Fig. 2. (a) An irregularly spaced set of points; (b) field surface obtained using inverse multiquadric basis functions with $r = 5$; (c) digitization of surface (b) into a 256×256 grid; (d) same as (c) except that elevations are shown in intensities; (e) contours representing minor ridges; (f) contours representing major ridges; (g) ridge contours; (h) RaG curves with $\sigma = 0.04$ approximating points in (a); (i) overlaying of (a) and (h).

be determined will change also. Next, we investigate the effect of parameter r on the obtained ridge contours.

5. EFFECT OF PARAMETER r ON RIDGE CONTOURS

As parameter r is varied, the shapes of the basis functions of the surface change. By increasing r , the basis functions become wider, resulting in a Fourier transform whose high spatial frequency coefficients are smaller

[Castleman 1996]. Since formula (8) shows a sum of these basis functions, as parameter r is increased, the surface will produce a Fourier transform whose high spatial frequency coefficients are smaller. This, in turn, means that as parameter r is increased, a smoother surface is obtained. Since smoother surfaces have smoother ridges, we can expect that the obtained ridge contours will be smoother also. Therefore, as parameter r is increased, smoother ridge contours are obtained.

Figures 3(a) to 3(d) show surface g when r is set to 3, 4.5, 6, and 10, respectively. The ridge contours of these surfaces are shown in Figures 3(e) to 3(h), respectively. Since parameter r characterizes the spatial frequency content or the smoothness of a field surface and its ridge contours, by selecting r appropriately, the contours can be made to have a desired smoothness. In Figure 3, we see that as r is increased, the contours become smoother, and as r is decreased, the contours become more detailed.

6. NODE ESTIMATION

The method outlined above determines the ridge contours as approximations to the curves to be found. These contours are used to partition a point set into subsets and order the points in each subset.

Suppose a point set has produced m ridge contours; then a point is assigned to contour j ($1 \leq j \leq m$) if it is closer to a pixel in contour j than to a pixel in any other contour. In this manner, a point is assigned to one of m ridge contours. This process when completed will partition a point set into m subsets by assigning the points into one of m ridge contours. Figures 4(a) and 4(b) show the point subsets obtained in this manner from the point set of Figure 2(a).

To order points $\{\mathbf{q}_i : i = 1, \dots, n\}$ in subset j , for each point \mathbf{q}_i , a pixel that is closest to it in contour j is determined. We call the obtained pixel the *projection* of point \mathbf{q}_i onto the contour. After determining projections of all points in the subset to the contour, the contour is traced from one end to the other, and the associated points are ordered in the order in which the projections are visited. To trace an open contour, one of its end pixels is taken as the start pixel and at each step, the pixel adjacent to the start pixel is located by examining its eight neighbors. Since a start pixel has only one neighbor, the next pixel on the contour is uniquely determined. Then the start pixel is marked as the background and the newly found pixel is marked as the start pixel. The process is repeated until the next pixel found is the end pixel. If a contour is closed, one of its pixels is taken as the end pixel, and the next pixel to be visited is taken as the start pixel. Then the contour is traced just like an open contour.

Since the ridge contours are approximations to the curves to be found, the contour length from a projection to the start of the contour is divided by the length of the contour to obtain the node at the point. Nodes estimated in this manner are proportional to contour lengths or arc lengths of points on the curve to a start point. If the contour is open, one of the end pixels is

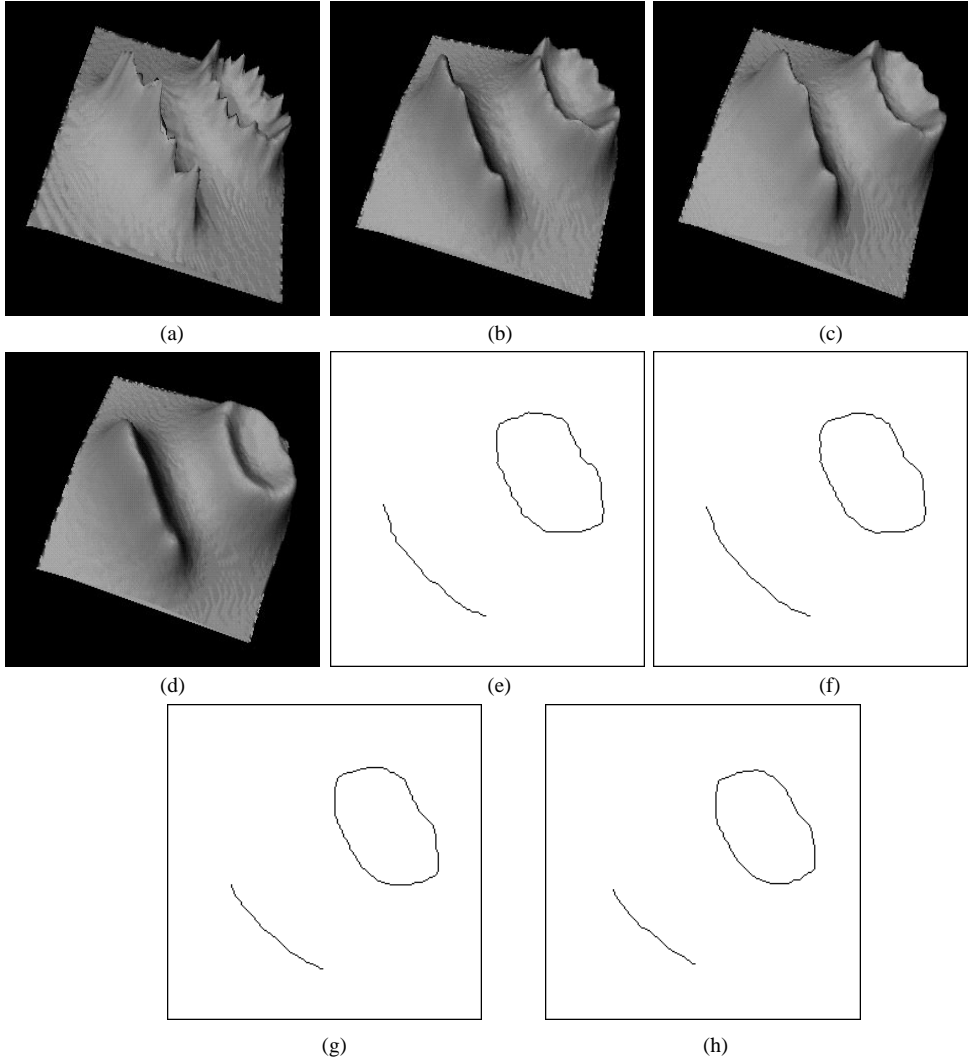


Fig. 3. (a) to (d); field surfaces obtained when setting $r = 3, 4.5, 6,$ and 10 , respectively; (e) to (h); ridge contours of surfaces (a) to (d), respectively.

taken as the start pixel. If the contour is closed, an arbitrary pixel on the contour is taken as the start pixel.

The size of the image obtained by digitizing surface g determines the accuracy of the obtained nodes. If surface g is very coarsely digitized, the obtained ridge contours will be very short and numerous points may produce the same node, especially when given points are dense. To provide a more accurate node estimation, surface g should be digitized into a larger image.

Once the coordinates of given points and the associated nodes are known, a parametric curve can be fitted to the points by one of the existing

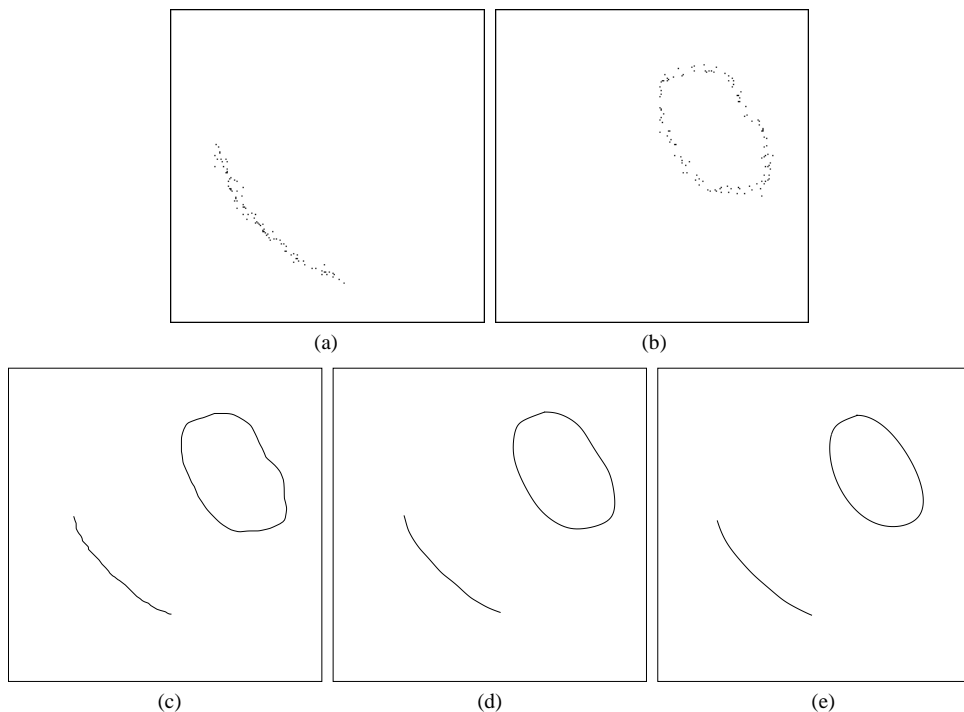


Fig. 4. (a) and (b); two point subsets obtained from the points in a; (c) to (e) RaG curves with $\sigma = 0.01, 0.04,$ and $0.08,$ respectively, fitting the two point subsets.

methods [Goshtasby 1995; Grossman 1970; Hoschek 1987; 1988b; Piegl 1987; Speer et al. 1998]. Fitting rational Gaussian or RaG curves [Goshtasby 1995] to the points in Figure 2(a) with nodes as determined above, we obtain the curves shown in Figure 2(h). The curves are overlaid with the original points in Figure 2(i) to show the quality of curve fitting. Note that these curves were obtained using the points in Figure 2(a) and not the contour pixels in Figure 2(g). The contour pixels were used only to partition a point set into subsets and determine the nodes of the points.

The RaG formulation was selected over B -splines and other parametric formulations because the standard deviation of Gaussians in the RaG formulation can be used as a smoothness parameter to produce curves with different smoothness fitting the same set of points. To achieve a similar effect in B -splines, it is necessary to change the degree of the curve. Figures 4(c) to 4(e) show RaG curves with different standard deviations (smoothness levels) fitting the points in Figure 2(a). At a small standard deviation, obtained curves are detailed, and as the standard deviation is increased the curves become smoother. If a very large standard deviation is selected, as shown in Figure 4(e), the curves may no longer be able to trace the spines of the points as they have to satisfy a very high smoothness constraint. The smoothness of the curves should be selected either interactively or from the information available about the shapes to be recovered.

7. CHOICE OF BASIS FUNCTIONS

The process described above for curve fitting used inverse multiquadrics to obtain the field surface and, by tracing the ridges of the surface, determined approximations to the curves. The same process can be followed using other monotonically decreasing radial basis functions such as Gaussians. This process is actually not limited to monotonically decreasing basis functions. Monotonically increasing basis functions, such as multiquadric and logarithmic, can be used also, but then, instead of the ridges, the troughs of the obtained field surface should be traced to obtain the contours. In this section, we present the curve-fitting results when Gaussian basis functions are used to represent the field surface.

The points in Figure 2(a) are shown again in Figure 5(a). Centering a Gaussian field of unit height and standard deviation 5 at each point and adding the fields, we obtain the surface depicted in Figure 5(b). We see that, compared to Figure 2(b), the field surface obtained from the Gaussian bases decreases more rapidly from the given points and approaches zero faster. The surface after digitization is shown in Figure 5(c) and again in 5(d) in intensity form. Since the surface obtained from the Gaussian bases approaches zero faster than the surface obtained from inverse multiquadrics, the minor ridges of the surface will contain shorter branches. Interestingly, although the numbers and the positions of branches in minor ridges obtained by inverse multiquadrics and Gaussians are different, and noticeable differences exist between the major ridges in field surfaces obtained by inverse multiquadrics and Gaussians, there is very little difference between ridge contours obtained by the two basis functions. Since the contours obtained using inverse multiquadrics and Gaussians are very similar, both basis functions partition the points into similar subsets. In addition, since the contours from the two basis functions are very similar, the nodes calculated using the contours are very similar. As a result, the curves obtained from the two basis functions will be almost identical, as can be observed by comparing Figures 2(h) and 5(h).

By increasing the standard deviation of the Gaussian bases, the ridges of the field surface become smoother, as can be observed in Figures 6(a) to (d). The ridge contours obtained, as shown in Figures 6(e) to (h), are very similar to those obtained by inverse multiquadrics. After partitioning the point set into two subsets (see Figures 7(a) and 7(b)) and fitting a RaG curve to each subset, we obtain the curves depicted in Figures 7(c)–(e). The standard deviation of RaG curves in these figures are 0.01, 0.04, and 0.08, respectively. Note that the standard deviation of Gaussians used in RaG curves does not have a unit, but the standard deviation of Gaussians used to obtain the field surface is measured in pixels. The curves obtained by multiquadrics and Gaussian bases have very small differences due to small differences in the ridge contours obtained by the two field surfaces.

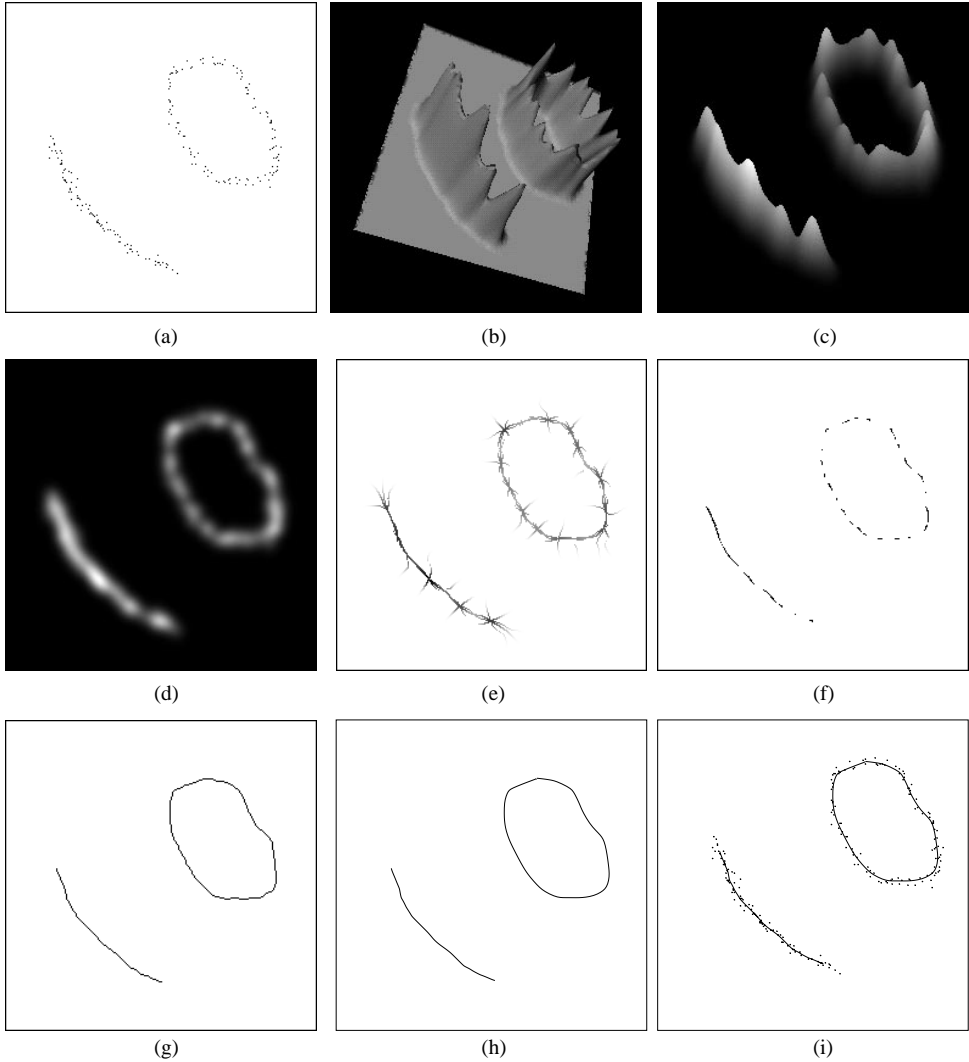


Fig. 5. (a) Same as points shown in 2(a); (b) field surface obtained when using Gaussians of standard deviation 5 pixels as basis functions; (c) digitization of surface (b) into a 256×256 grid; (d) same as (c) except that intensities show surface elevations; (e) minor ridges; (f) major ridges; (g) ridge contours; (h) RaG curves with $\sigma = 0.04$ fitting the points in (a); (i) overlaying of (a) and (h).

8. FURTHER OBSERVATIONS

To demonstrate the behavior of the proposed curve-fitting method, examples using four additional point sets are presented in Figure 8. Figure 8(a) shows noisy points along an open contour; Figure 8(d) shows a dense and noisy point set along the silhouette of a coffee mug; Figure 8(g) shows irregularly spaced points along the silhouette of a model plane and one of its wings; and Figure 8(j) shows an example of a data set containing two closed geometric structures. We can see the geometric structures in these

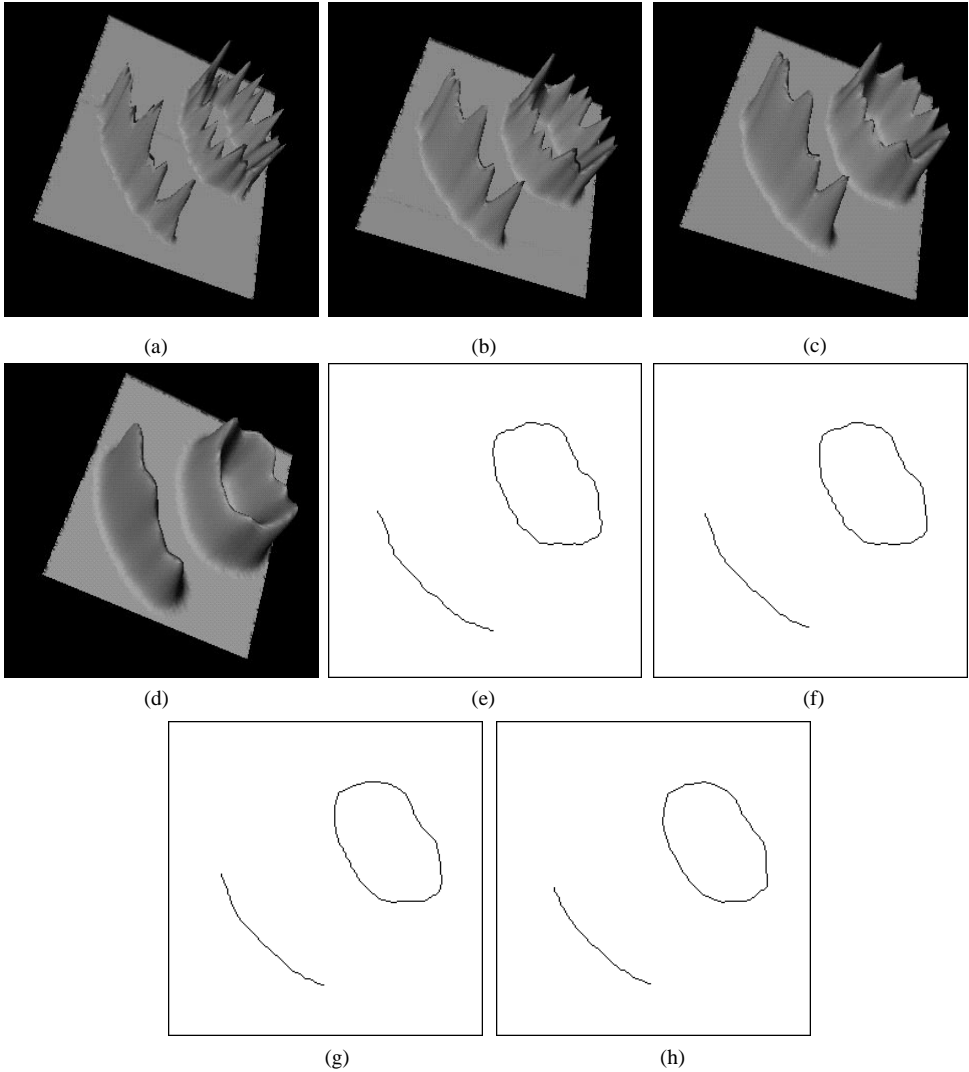


Fig. 6. (a) to (d); field surfaces obtained when setting the standard deviation of Gaussians to 3, 4.5, 6, and 10, respectively. (e) to (h); ridge contours of surfaces (a)—(d), respectively.

point sets and, if asked, can trace them manually without any difficulty. We would like to see if our algorithm can do the same.

The curves obtained using inverse multiquadrics are shown in the center column of Figure 8, while the curves obtained from Gaussians are shown in the right column of the figure. Parameters r of the inverse multiquadrics and σ of the Gaussians were both set to 5. As can be observed, corresponding curves by inverse multiquadrics and Gaussians are very similar. The main difference between them exists in Figures 8(b) and 8(c). Inverse multiquadrics have produced two ridge contours and therefore two curves, while the Gaussians have produced one ridge contour and therefore one

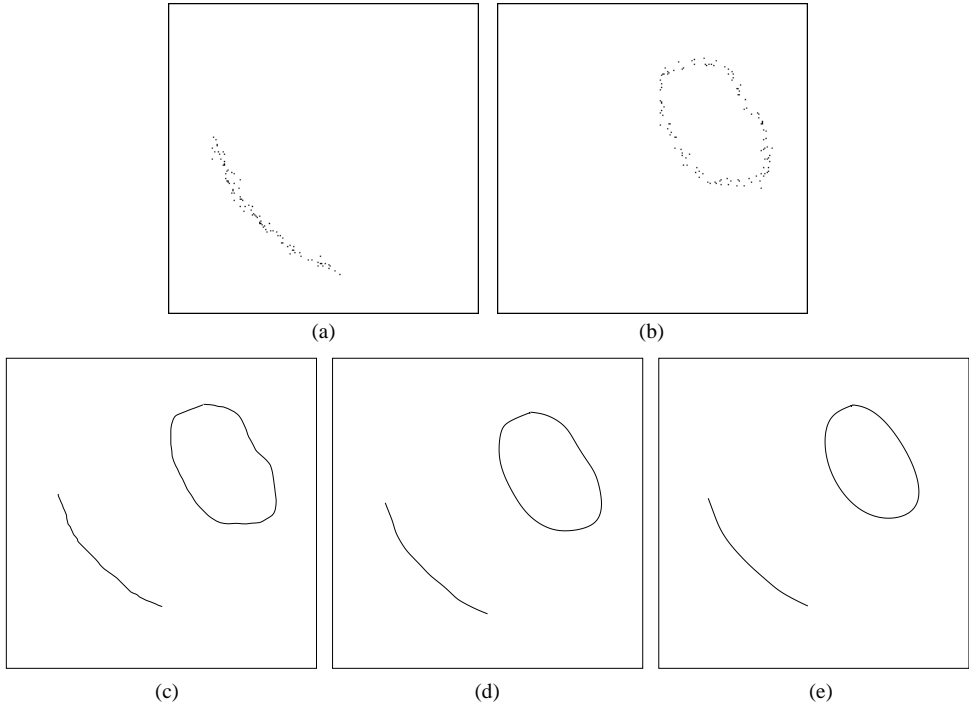


Fig. 7. (a), (b) Point subsets obtained from points in Figure 5(a) using contours of Figure 5(g). (c)-(e) RaG curves with $\sigma = 0.01, 0.04,$ and $0.08,$ respectively, fitting the point subsets.

curve. The field surfaces obtained by inverse multiquadrics and Gaussians are generally different; therefore, when their ridges are traced, there is no guarantee that they will produce the same number of contours. We have found that, in general, Gaussians produce longer and fewer contours than inverse multiquadrics.

The point sets shown in Figure 8 do not contain geometric structures with branches and loops. If a point set contains branches and loops, the ridge contours will also contain branches and loops. A single curve segment, however, cannot represent branching structures. The solution we propose is to partition the ridges at the branch points, combine adjoining ridge segments into ridge contours, and fit a curve to each ridge contour. An example of this is demonstrated in Figure 9. In our procedure, from among the different possibilities, two adjacent contour segments are joined in such a way that the longest contour is obtained. Using the point set shown in Figure 9(a), we obtained the ridge contours shown in Figure 9(b), and the curves shown in Figure 9(c). An alternative method for joining ridge segments at branch points is to take the two segments whose tangents at the branch point are the closest. If some information about the geometric structures to be recovered is available, that information should be used to select the ridge segments and construct the ridge contours. The curves in Figure 9(c) were obtained using Gaussian fields of standard deviation 5 pixels and RaG curves of standard deviation 0.04.

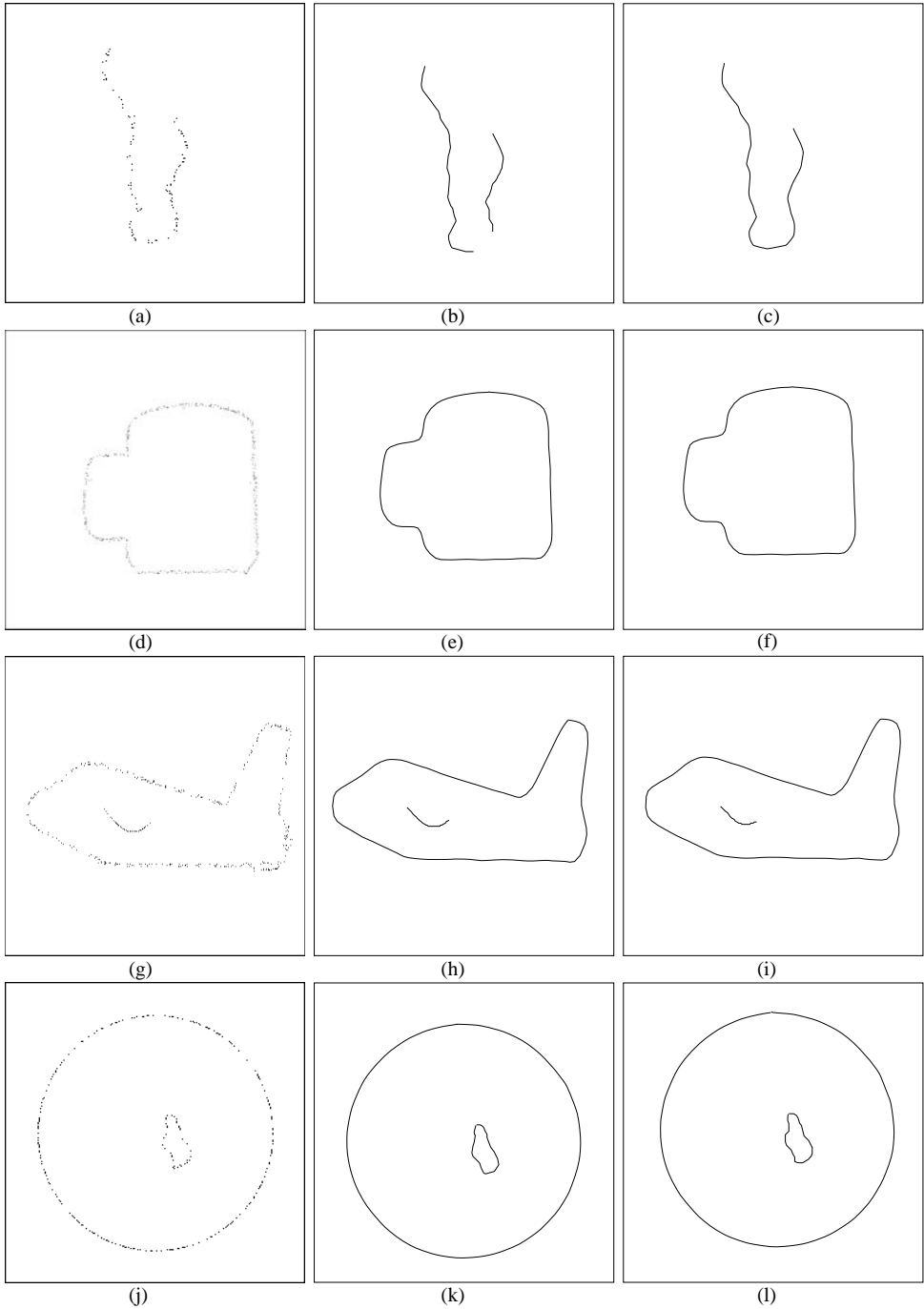


Fig. 8. Curve-fitting results for four additional point sets (left column) using inverse multiquadric (center column) and Gaussian (right column) basis functions.

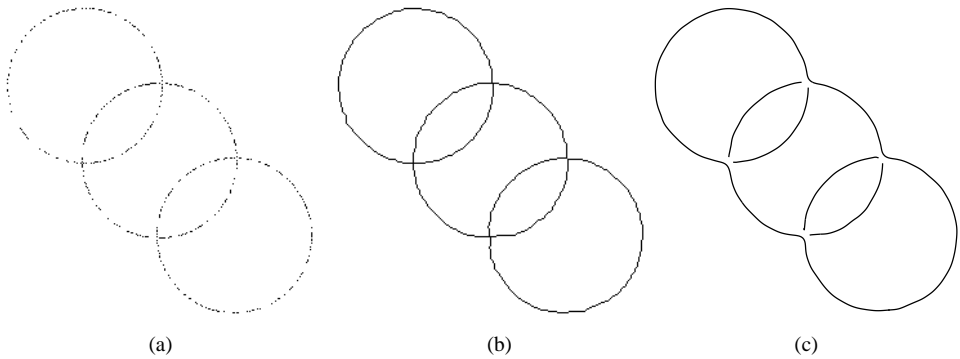


Fig. 9. (a) A point set containing loops and branches; (b) ridge segments obtained using Gaussian basis functions of standard deviation 5 pixels; adjoining ridge segments were combined to produce the longest ridge contours possible, and the contours were then used to partition the points into subsets; (c) RaG curves with $\sigma = 0.04$ fitting the point subsets.

9. SUMMARY AND CONCLUSIONS

A large number of techniques for fitting parametric curves to irregularly spaced points have been developed. These techniques fit a single curve to a set of points and often require that the adjacency relation between the points be given. In science and engineering problems that deal with measurement data, the adjacency relation between the points may not be available and the points may be noisy. Moreover, it may not be appropriate to fit a single curve segment to all the points. In this article a method to partition a point set into subsets and fit a parametric curve to each subset was described. The proposed method has the ability to take into consideration the noisiness and denseness of a point set when obtaining the curves.

Also introduced was a method to determine the nodes of a parametric curve that approximates a set of dense and noisy points. The proposed method provides the means to fit any parametric curve, including B -splines and nonuniform rational B -splines, to irregularly spaced points. Although in this article only inverse multiquadrics and Gaussians were used as basis functions to obtain a field surface from which the number of curve segments were determined, other radial basis functions [Powell 1987] may be used in the same manner to find the number of curves fitting a set of dense and noisy points. Depending on the parametric curve formulation and the radial basis functions used, the number and the shapes of obtained curves may vary.

ACKNOWLEDGMENTS

I would like to thank the National Science Foundation for partially funding this work under grant IIS-9906340, and also the anonymous reviewers for their insightful comments.

REFERENCES

- BANFIELD, D. B. AND RAFTERY, A. E. 1992. Ice floe identification in satellite images using mathematical morphology and clustering about principal curves. *J. Am. Stat. Assoc.* 87, 7–16.
- BEACH, R. C. 1991. *An Introduction to the Curves and Surfaces of Computer-Aided Design*. Van Nostrand Reinhold Co., New York, NY.
- BUHMANN, M. D. 1990. Multivariate cardinal interpolation with radial basis functions. *Constr. Approx.* 6, 225–255.
- CASTLEMAN, K. R. 1996. *Digital Image Processing*. Prentice Hall Press, Upper Saddle River, NJ.
- COHEN, E. AND O'DELL, C. L. 1989. A data dependent parametrization for spline approximation. In *Mathematical Methods in Computer Aided Geometric Design*, T. Lyche and L. L. Schumaker, Eds. Academic Press Prof., Inc., San Diego, CA, 155–166.
- DYN, N. 1987. Interpolation of scattered data by radial functions. In *Topics in Multivariate Approximation*. Academic Press, Inc., New York, NY, 47–61.
- FARIN, G. 1990. *Curves and Surfaces for Computer Aided Geometric Design*. Academic Press Prof., Inc., San Diego, CA.
- FAROUKI, R. T. 1997. Optimal parameterizations. *Comput. Aided Geom. Des.* 14, 2, 153–168.
- FAUX, I. D. AND PRATT, M. J. 1979. *Computational Geometry for Design and Manufacture*. Ellis Horwood, Upper Saddle River, NJ.
- LYCHE, T. AND SCHUMAKER, L. L., EDS. 1989. *Mathematical Methods in Computer Aided Geometric Design*. Academic Press Prof., Inc., San Diego, CA.
- FRITSCH, R. E. AND CARLSON, R. E. 1980. Monotone piecewise cubic interpolation. *SIAM J. Numer. Anal.* 17, 238–246.
- GOSHTASBY, A. 1995. Geometric modeling using rational gaussian curves and surfaces. *Comput. Aided Des.* 27, 363–375.
- GOSHTASBY, A. AND O'NEILL, W. D. 1993. Surface fitting to scattered data by a sum of Gaussians. *Comput. Aided Geom. Des.* 10, 2 (Apr.), 143–156.
- GROSSMAN, M. 1970. Parametric curve fitting. *Computer J.* 14, 169–172.
- HAGEN, H., ED. 1992. *Curve and Surface Design*. SIAM, Philadelphia, PA.
- HARDY, R. L. 1990. Theory and applications of the multiquadrics--biharmonic method. *Comput. Math. Appl.* 19, 163–208.
- HARTLEY, P. J. AND JUDD, C. J. 1980. Parametrization and shape of b-spline curves for cad. *Comput. Aided Des.* 12, 235–238.
- HASTIE, T. AND STUETZLE, W. 1989. Principal curves. *J. Am. Stat. Assoc.* 84, 502–516.
- HÖLZLE, G. E. 1983. Knot placement for piecewise polynomial approximation of curves. *Comput. Aided Des.* 15, 295–296.
- HOSCHEK, J. 1987. Approximate conversion of spline curves. *Comput. Aided Geom. Des.* 4, 59–66.
- HOSCHEK, J. 1988. Intrinsic parametrization for approximation. *Comput. Aided Geom. Des.* 5, 27–31.
- HOSCHEK, J. 1988. Spline approximation of offset curves. *Comput. Aided Geom. Des.* 5, 1 (June), 33–40.
- ICHIDA, K. AND KIYONO, T. 1977. Curve fitting by a one-pass method with a piecewise cubic polynomial. *ACM Trans. Math. Softw.* 3, 164–174.
- JÜTTLER, B. 1997. A vegetarian approach to optimal parameterizations. *Comput. Aided Geom. Des.* 14, 9, 887–890.
- KANSA, E. J. 1990. Multiquadrics: A scattered data approximation scheme with applications to computational fluid dynamics-I. *Comput. Math. Appl.* 19, 127–145.
- KOSTERS, M. 1991. Curvature-dependent parametrization of curves and surfaces. *Comput. Aided Des.* 23, 8, 569–578.
- LANCASTER, P. AND SALKAUSKAS, L. 1986. *Curve and Surface Fitting*. Academic Press, Inc., New York, NY.
- LEE, E. 1989. Choosing nodes in parametric curve interpolation. *Comput. Aided Des.* 6, 363–370.

- LEE, I. -K. 2000. Curve reconstruction from unorganized points. *Comput. Aided Geom. Des.* 17, 161–177.
- LEVIN, D. 1998. The approximation power of moving least-squares. *Math. Comput.* 67, 224, 1517–1531.
- MARIN, S. P. 1984. An approach to data parametrization in parametric cubic spline interpolation problem. *J. Approx. Theory* 41, 1 (May), 64–86.
- MEINGUET, J. 1979. An intrinsic approach to multivariate spline interpolation at arbitrary points. In *Polynomial and Spline Approximation*, B. N. Sahney, Ed. D. Reidel Publishing Co., Inc., New York, NY, 163–190.
- MULLINEUX, M. 1982. Approximating shapes using parametrized curves. *IMA J. Appl. Math.* 29, 203–220.
- PIEGL, L. 1987. A technique for smoothing scattered data with conic sections. *Comput. Ind.* 9, 3 (Nov.), 223–238.
- POTTMANN, H. AND RANDRUP, T. 1998. Rotational and helical surface approximation for reverse engineering. *Computing* 60, 4, 307–322.
- POWELL, M. J. D. 1987. Radial basis functions for multivariable interpolation: a review. In *Algorithms for approximation*, J. C. Mason and M. G. Cox, Eds. Clarendon Press Institute of Mathematics and its Applications conference series. Clarendon Press, New York, NY, 143–167.
- SARKAR, B. AND MENQ, C.-H. 1991. Parameter optimization in approximating curves and surfaces to measurement data. *Comput. Aided Geom. Des.* 8, 4 (Oct.), 267–290.
- SCHAGEN, I. P. 1980. The use of stochastic processes in interpolation and approximation. *Int. J. Comput. Math.*, 63–76.
- SPEER, T., KUPPE, M., AND HOSCHEK, J. 1998. Global reparametrization for curve approximation. *Comput. Aided Geom. Des.* 15, 9, 869–877.

Received: June 1998; revised: January 2000; accepted: April 2000