Visual Comput (2005)
DOI 10.1007/s00371-005-0340-0

ORIGINAL ARTICLE

Zhouwang Yang
Jiansong Deng
Falai Chen

Fitting unorganized point clouds with active implicit B-spline curves

Published online: 31 August 2005

Z. Yang (✉) · J. Deng · F. Chen
Department of Mathematics
University of Science and Technology of China
Hefei, Anhui 230026, P.R. China
yangzw@ustc.edu.cn

Abstract In computer-aided geometric design and computer graphics, fitting point clouds with a smooth curve (known as curve reconstruction) is a widely investigated problem. In this paper, we propose an active model to solve the curve reconstruction problem, where the point clouds are approximated by an implicit B-spline curve, i.e., the zero set of a bivariate tensor-product B-spline function. We minimize the geometric distance between the point clouds and the implicit B-spline curve and an energy term (or smooth term) which helps to extrude the possible extra branches of the implicit curve. In each step of the iteration, the trust region algorithm in optimization theory is applied to solve the corresponding minimization problem. We also discuss the proper choice of the initial shape of the approximation curve. Examples are provided to illustrate the effectiveness and robustness of our algorithm. The examples show that the proposed algorithm is capable of handling point clouds with complicated topologies.

Keywords Curve reconstruction · Active implicit B-spline curve · Trust region algorithm · Geometric distance

1 Introduction

With the development of modern industry, higher surface quality and aesthetic requirements of CAM products are being increasingly considered in many industries, such as jewelry and automobile industries. Free-form curves and surfaces are used in such products. However, there are difficulties in designing 3D free-form surfaces on 2D computer screens using traditional CAD systems. Hence, physical media such as clay models are first designed. Meanwhile, some existing models and products with complex free-form surfaces need to be reproduced, for instance, some valuable antiques, models of human brains, bones, teeth, etc., need to be precisely reconstructed for research or medical treatment. The task of converting a 3D prototype object into a computer model for subsequent CAD/CAM processes, is then left to the computer system and specialized software.

During the fulfillment of the task, as the first step, point clouds are generated with the coordinates of each point captured from the surfaces of the existing models and products. Some of these data, especially, the image contours from the medical field, show curve shapes need to be fitted with smooth curves. Here the data points are unorganized, non-uniformly distributed, and possibly with noise. The corresponding problem is called curve reconstruction, where the curves are usually represented in their parametric forms or implicit forms. Curve reconstruction has been widely studied in computer graphics and geometric modeling in the past decade. It has various applications in CAD/CAM, computer vision, and many other disciplines. A main approach to curve reconstruction is based on the least-square fitting with a regularization term that represents the fairness of the final result. Let $\boldsymbol{C}(t)$ be a parametric curve, which is a linear combination of some basis functions, such as the B-spline bases, the radial-basis functions, etc. Then, the approximation curve is computed by minimizing the objective function $R = \mathrm{dist}(\{\boldsymbol{P}_i\}_{i=1}^{M}, \boldsymbol{C}(t)) + wT_s$, where $\{\boldsymbol{P}_i\}_{i=1}^{M}$ are given data points, T_s is the regulation term and w is a weight. In general, both the first term and the second term are the quadratic functions in the unknown control points $\boldsymbol{C}_j$. Hence minimization of R leads to the solution of a linear system of equations. The solution depends largely on the proper parameterization of the data set. However, it is a difficult problem to estimate a proper parameter value t_i corresponding to a point $\boldsymbol{P}_i$ of the unorganized data, though some data parameterization techniques have been proposed [11, 15].

Another approach in parametric curve reconstruction is based on the active contour models, also called *snakes*, which are proposed by Kass et al. [13]. This technique is originally introduced for detecting image contours, and later is extended to other areas such as computer vision [2]. Recently, Pottmann et al. [18] applied the technique to curve approximation, and they proposed an active parametric B-spline model to fit unorganized points [19]. Wang et al. [22] explore the above idea more thoroughly and improve the algorithm dramatically.

One problem with parametric curve fitting methods is that it is difficult to handle data sets with complicated topology. Furthermore, as we addressed earlier, parameterization is always a non-trivial problem in curve fitting. To overcome these difficulties, implicit representations for curves/surfaces are introduced.

The common approach in implicit curve/surface reconstruction uses a combination of some smooth basis functions, say radial basis functions, to find a scalar function such that all data points are close to an iso-contour of that scalar function [3, 16, 17, 21]. In [6] polyharmonic radial-basis functions and multi-pole methods are introduced, enabling the authors to model large data sets by a single radial basis function. The signed distance function has been used to reconstruct and represent an implicit curve/surface on a rectangular grid with the signs to distinguish inside and outside [1, 4, 10]. Similar ideas have been applied to shape reconstruction from range data and image fusion [7, 9]. Zhao et. al. [24, 25] applied the level set method in surface reconstruction by solving a PDE equation numerically; Jüttler [12] described a technique for fitting implicitly defined algebraic spline curves and surfaces to scattered data by simultaneously approximating points and associated normal vectors which are estimated from the given data. In all these methods, either additional information such as normals, signed distance functions has to be provided, or the algorithm is inefficient and/or not robust.

Hence, a good curve reconstruction algorithm should be able to deal with data sets with complicated topology as well as with noise and non-uniformity, without requiring some additional information, which is not easy to obtain in some circumstances. The final result should be a reasonable shape with good approximation to the data set. In this paper, we present an active implicit B-spline model, similar to the active parametric B-spline model proposed by Pottmann [19], to solve the curve reconstruction problem. In this model, we start with some properly specified initial shape for the active implicit B-spline curve, then we iteratively modify the active curve such that it converges to the target shape of the data points by solving some optimization problem. The algorithm stops after the implicit curve well approximates the data set. In this model, only the positions of data points are taken as input.

The rest of the paper is organized as follows. In Sect. 2, the implicit B-spline curve is introduced. In Sect. 3, we study the local approximation of the geometric distance function and setup the building block for the active model. In Sect. 4, the active model is proposed, and the reconstruction scheme with active implicit B-spline curves is outlined. The trust region algorithm is applied to solve the corresponding optimization problem in the reconstruction. In Sect. 5, some examples are implemented and high-quality reconstruction curves are illustrated. Finally, in Sect. 6, we conclude the paper with some problems for future research.

2 Algebraic tensor-product B-spline curves

Let $f(x, y)$ (or $f(\boldsymbol{P})$) be a bivariate tensor-product B-spline function of bi-degree (l, l') defined over some domain Ω:

$$f(x, y) = \sum_{r,s} c_{rs} M_r(x) N_s(y) , \tag{1}$$

where $\{M_r(x)\}_{r=1}^{m}$ and $\{N_s(y)\}_{s=1}^{n}$ denote the B-spline basis functions of degree l and l' with some given knot sequences. The zero set of the function f is defined by

$$V(f) = \{(x, y) \in \Omega \subset \mathbb{R}^2 \mid f(x, y) = 0\} , \tag{2}$$

and it is called an *implicit B-spline curve*. For a fixed set of basis functions, the implicit B-spline curve is determined by the coefficients $\{c_{rs}\}_{m\times n}$ (called the *control coefficients*). For simplicity of notations, the control coefficients and the basis functions are gathered (in a suitable ordering) into two column vectors, denoted by $\boldsymbol{f}$ and $\boldsymbol{q}(\boldsymbol{P})$ (or simply $\boldsymbol{q}$), respectively. Using the notations, we can write $f(\boldsymbol{P})$ as

$$f(\boldsymbol{P}) = \boldsymbol{q}(\boldsymbol{P})^{\tau} \boldsymbol{f} = \boldsymbol{q}^{\tau} \boldsymbol{f} , \tag{3}$$

and the gradient of $f(\boldsymbol{P})$ as

$$\nabla f(\boldsymbol{P}) = \left(\frac{\partial f}{\partial x}(\boldsymbol{P}), \frac{\partial f}{\partial y}(\boldsymbol{P}) \right)^{\tau} = \begin{pmatrix} \boldsymbol{u}^{\tau} \boldsymbol{f} \\ \boldsymbol{v}^{\tau} \boldsymbol{f} \end{pmatrix} , \tag{4}$$

where $\boldsymbol{u} = \frac{\partial \boldsymbol{q}}{\partial x}$ and $\boldsymbol{v} = \frac{\partial \boldsymbol{q}}{\partial y}$.

The curve reconstruction problem is to find a B-spline function f such that $V(f)$ gives a good approximation to the given point clouds. We require that the geometric distance between the implicit curve $V(f)$ and the point clouds be as small as possible. On the other hand, we hope the implicit curve has a good "quality", such as continuity and fairness. In this paper, we impose the condition that the implicit curve has a minimal simplified thin-plate energy [5]:

$$\mathrm{Eng}_{\mathrm{T}}(\boldsymbol{f}) = \iint_{\Omega} \left(f_{xx}^2 + 2 f_{xy}^2 + f_{yy}^2 \right) dxdy = \boldsymbol{f}^\tau \boldsymbol{H} \boldsymbol{f}. \quad (5)$$

It is quadratic in the coefficient vector $\boldsymbol{f}$, and the symmetrical matrix $\boldsymbol{H}$ can be computed by a Gauss integral.

3 Local approximation of geometric distance

The active model of the implicit curve reconstruction we propose in the next section heavily relies on a local approximation of the geometric distance of a given data point to an active implicit curve.

Let $\{\boldsymbol{P}_i\}_{i=1}^M$ be a collection of unorganized data points. At first, one needs to define a meaningful metric

$$\mathrm{Err}(\boldsymbol{f}) = \mathrm{Err}(\{\boldsymbol{P}_i\}_{i=1}^M, V(f)) , \quad (6)$$

as the error function of the data set $\{\boldsymbol{P}_i\}_{i=1}^M$ to an implicit curve $V(f)$. Generally, the geometric distance is an optimal choice for this metric. For every point $\boldsymbol{P}$ in the data set, the geometric distance $d(\boldsymbol{P}, V(f))$ is the Euclidean distance from $\boldsymbol{P}$ to the implicit curve $V(f)$:

$$d(\boldsymbol{P}, V(f)) = \min_{\boldsymbol{Y} \in V(f)} \|\boldsymbol{P} - \boldsymbol{Y}\| . \quad (7)$$

Suppose $\boldsymbol{X} \in V(f)$ is the nearest point (foot-point) on the implicit curve to $\boldsymbol{P}$. It then satisfies the following preconditions:

$$\begin{cases} f(\boldsymbol{X}) = 0 , \\ (\boldsymbol{P} - \boldsymbol{X})^\tau \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \nabla f(\boldsymbol{X}) = 0 . \end{cases} \quad (8)$$

The error function is defined as the squared sum of the geometric distances:

$$\mathrm{Err}_{\mathrm{g}}(\boldsymbol{f}) = \sum_{i=1}^{M} d^2(\boldsymbol{P}_i, V(f)) . \quad (9)$$

An ideal algorithm to curve reconstruction would minimize the geometric distance error Eq. 9. Unfortunately, this metric results in an intractable minimization problem whose solution cannot be expressed analytically in closed form because of the non-linearity of $V(f)$. To avoid the difficulty of numerical minimization, we consider a local approximation of the geometric distance $d(\boldsymbol{P}, V(f))$, with the unknown coefficients vector $\boldsymbol{f}$ being the variables.

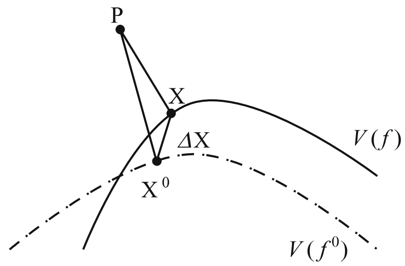


Fig. 1. Estimation of foot-point on an active implicit curve

Suppose $V(f^0)$ is an implicit B-spline curve computed in some iterative step, and $\boldsymbol{P}$ is a point in the point clouds. Let $\boldsymbol{f}^0$ denote the coefficient vector of f^0 and $\boldsymbol{X}^0 = \boldsymbol{X}(f^0)$ be the foot-point of $\boldsymbol{P}$. The foot-point $\boldsymbol{X}^0$ and the geometric distance $d(\boldsymbol{P}, V(f^0)) = \|\boldsymbol{P} - \boldsymbol{X}^0\|$ can be computed numerically using the technique called the nearpoint procedure [8].

After one iteration, suppose the control coefficients vector is changed to $\boldsymbol{f} = \boldsymbol{f}^0 + \boldsymbol{g}$. We attempt to estimate the new foot-point $\boldsymbol{X}(f)$ on the implicit curve $V(f)$ (see Fig. 1). It is easy to see that

$$\boldsymbol{X}(f) \approx \boldsymbol{X}(f^0) + \boldsymbol{L}[f^0]^\tau \boldsymbol{g} = \boldsymbol{X}^0 + \Delta \boldsymbol{X} , \quad (10)$$

where $\boldsymbol{L}[f^0] = (\boldsymbol{l}_x[f^0], \boldsymbol{l}_y[f^0])$ is the coefficients of the linear part in the Taylor expansion of $\boldsymbol{X}(f)$ at f_0.

We expand Eq. 8 at $\boldsymbol{f}^0$ and obtain

$$[\nabla f^0(\boldsymbol{X}^0) + \nabla g(\boldsymbol{X}^0)]^\tau \Delta \boldsymbol{X} = -g(\boldsymbol{X}^0), \quad (11)$$

$$\Big[(\boldsymbol{P} - \boldsymbol{X}^0)^\tau \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \left(\nabla^2 f^0(\boldsymbol{X}^0) + \nabla^2 g(\boldsymbol{X}^0) \right) + \left(\nabla f^0(\boldsymbol{X}^0) + \nabla g(\boldsymbol{X}^0) \right)^\tau \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \Big] \Delta \boldsymbol{X} = -(\boldsymbol{P} - \boldsymbol{X}^0)^\tau \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \nabla g(\boldsymbol{X}^0) . \quad (12)$$

Solving the above linear system of equations about $\Delta \boldsymbol{X}$ gives

$$\Delta \boldsymbol{X} = \boldsymbol{L}[f^0]^\tau \boldsymbol{g} = \begin{pmatrix} \boldsymbol{l}_x[f^0]^\tau \boldsymbol{g} \\ \boldsymbol{l}_y[f^0]^\tau \boldsymbol{g} \end{pmatrix} . \quad (13)$$

Denote

$$\boldsymbol{P} - \boldsymbol{X}^0 = (a_1, a_2)^\tau, \quad \nabla f^0(\boldsymbol{X}^0) = (b_1, b_2)^\tau$$

$$\nabla^2 f^0(\boldsymbol{X}^0) = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix},$$

$$g(\boldsymbol{X}^0) = \boldsymbol{q}^\tau \boldsymbol{g},\ \nabla g(\boldsymbol{X}^0) = (\boldsymbol{u}^\tau \boldsymbol{g},\ \boldsymbol{v}^\tau \boldsymbol{g})^\tau,$$
$$\nabla^2 g(\boldsymbol{X}^0) = \begin{pmatrix} \boldsymbol{R}^\tau \boldsymbol{g} & \boldsymbol{S}^\tau \boldsymbol{g} \\ \boldsymbol{S}^\tau \boldsymbol{g} & \boldsymbol{T}^\tau \boldsymbol{g} \end{pmatrix},$$
$$\det[f^0] = \det \begin{pmatrix} b_1 & b_2 \\ \lambda_1 & \lambda_2 \end{pmatrix},$$

where $\lambda_1 = a_1c_{21} - a_2c_{11} - b_2$, $\lambda_2 = a_1c_{22} - a_2c_{12} - b_1$. It follows that

$$\begin{cases} \boldsymbol{l}_x[f^0] = \dfrac{b_2(a_1\boldsymbol{v} - a_2\boldsymbol{u}) - (a_1c_{22} - a_2c_{12} + b_1)\boldsymbol{q}}{\det[f^0]} \\ \boldsymbol{l}_y[f^0] = \dfrac{b_1(a_2\boldsymbol{u} - a_1\boldsymbol{v}) + (a_1c_{21} - a_2c_{11} - b_2)\boldsymbol{q}}{\det[f^0]} \end{cases}. \tag{14}$$

Hence the local approximation of the squared geometric distance $d^2(\boldsymbol{P}, V(f))$ is a quadratic function in $\boldsymbol{g}$:

$$\begin{aligned} &F(\boldsymbol{g};\ \boldsymbol{P},\ f^0) \\ &= \|\boldsymbol{P} - \boldsymbol{X}^0 - \boldsymbol{L}[f^0]^\tau \boldsymbol{g}\|^2 \\ &= \|\boldsymbol{P} - \boldsymbol{X}^0\|^2 - 2(\boldsymbol{P} - \boldsymbol{X}^0)^\tau \boldsymbol{L}[f^0]^\tau \boldsymbol{g} + \|\boldsymbol{L}[f^0]^\tau \boldsymbol{g}\|^2 . \end{aligned} \tag{15}$$

This is a building block for our new technique of implicit curve reconstruction presented in the next section.

4 Active model of implicit curve reconstruction

In this section, we propose a model for curve reconstruction with an implicit tensor-product B-spline curve $V(f)$. The order and the knot vectors of the B-spline basis functions are specified by users. Then the curve reconstruction problem can be formulated in the following optimization problem:

$$\min R(\boldsymbol{f}) = \sum_{i=1}^{M} d^2(\boldsymbol{P}_i,\ V(f)) + w\mathrm{Eng}_\mathrm{T}(\boldsymbol{f}) . \tag{16}$$

The first part in Eq. 16 is the geometric distance error from the data points to the implicit curve. The second part is a fairing term, and is applied to enforce the fairness of the final approximating curve. As we have stated in Sect. 3, it is a difficult task to estimate the geometric distance $d(\boldsymbol{P}_i, V(f))$ in an explicit form. An iterative correction procedure is proposed to overcome the difficulty based on the local approximation of geometric distance function.

4.1 An active model

The active implicit curve we are using is an algebraic spline curve, which is governed by its coefficients vector. The key idea to the active model is iteratively correcting the coefficient vector $\boldsymbol{f}$ with the help of the local approximation of the geometric distance function, so that the active implicit curve $V(f)$ deforms towards the target shape. In each step, we solve a minimization subproblem, which ensures that the geometric distance error function $\mathrm{Err_g}(\{\boldsymbol{P}_i\}_{i=1}^M,\ V(f))$ decreases quickly.

Our active model of the implicit curve reconstruction is outlined in the following steps:

1. Initialize the active implicit B-spline curve $V(f^0)$. Generally, we require that $V(f^0)$ encloses the input data points $\{\boldsymbol{P}_i\}_{i=1}^M$.
2. Repeatedly apply the following steps (a–c) until the approximation error reaches a pre-defined threshold or some stopping criterion is satisfied:
   (a) With the current coefficient vector $\boldsymbol{f}^k$, compute, for $i = 1, \ldots, M$, the foot-point $\boldsymbol{X}_i^k = \boldsymbol{X}_i(f^k)$ of $\boldsymbol{P}_i$ to $V(f^k)$ and $\boldsymbol{L}_i[f^k]$ based on Eq. 14.
   (b) Compute the displacement vector $\boldsymbol{g} = \boldsymbol{g}^k$ for the coefficient vector $\boldsymbol{f}^k$ by minimizing the function:
   $$Q^{(k)}(\boldsymbol{g}) = \sum_i F(\boldsymbol{g};\ \boldsymbol{P}_i,\ f^k) + w(\boldsymbol{f}^k + \boldsymbol{g})^\tau \boldsymbol{H}(\boldsymbol{f}^k + \boldsymbol{g}).$$
   (c) Set $\boldsymbol{f}^{k+1} = \boldsymbol{f}^k + \boldsymbol{g}^k$.
3. Output the active implicit curve $V(f^{k+1})$ as the final approximation to the target shape model.

Figure 2 shows one step in the active model of implicit curve reconstruction.

It should be noted that $Q^{(k)}(\boldsymbol{g})$ is only a local quadratic approximation of the original function $R(\boldsymbol{f}^k + \boldsymbol{g})$. In a certain iteration step, $\boldsymbol{g}^k$ may not necessarily make the objective function decrease. We will apply the trust region technique to obtain global convergence in the next section.

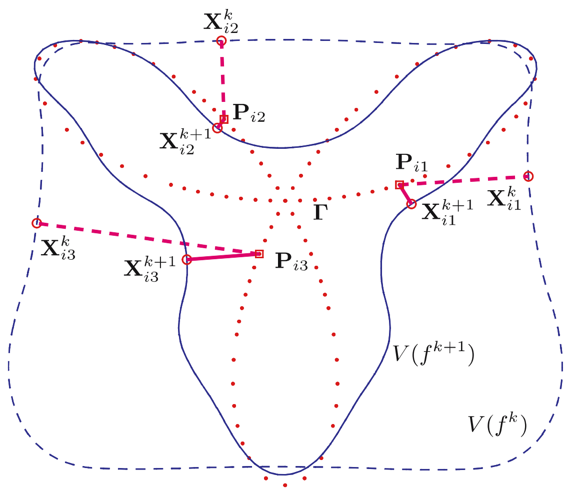


Fig. 2. One step in the active implicit curve reconstruction

4.2 Trust region algorithms

Trust region algorithms in optimization theory are a group of algorithms for ensuring global convergence while retaining fast local convergence. Many different versions of trust region algorithms have been proposed and applied to minimization problem in practice [20]. In all these versions, we first choose a trial step length e_k and then select the best step of this length by solving the following quadratic problem:

$$\begin{aligned} \min \quad & Q^{(k)}(\boldsymbol{g}) \\ s.t. \quad & \boldsymbol{g}^\tau \boldsymbol{g} \leqslant e_k^2 \,. \end{aligned} \tag{17}$$

The trial step length e_k is ensured an estimation of how far we trust the quadratic model, and is called a trust radius.

In this section, we will consider the curve reconstruction problem Eq. 16 by utilizing the trust region technique. Denote $r_i(\boldsymbol{f}) = \|\boldsymbol{P}_i - \boldsymbol{X}_i(\boldsymbol{f})\|$ and $r(\boldsymbol{f}) = (r_1(\boldsymbol{f}), \cdots, r_M(\boldsymbol{f}))^\tau$. We reformulate the problem of the implicit curve reconstruction as a non-linear optimization problem:

$$\min R(\boldsymbol{f}) = \frac{1}{2}\left[r(\boldsymbol{f})^\tau r(\boldsymbol{f}) + w\mathrm{Eng}_\mathrm{T}(\boldsymbol{f})\right] \,. \tag{18}$$

At the kth iteration, if $\boldsymbol{f}^k$ does not satisfy the Kunh–Tucker conditions, we calculate a trail step by solving the trust-region subproblem:

$$\begin{aligned} \min \quad & Q^{(k)}(\boldsymbol{g}) = R(\boldsymbol{f}^k) + \boldsymbol{b}_k^\tau \boldsymbol{g} + \frac{1}{2}\boldsymbol{g}^\tau \boldsymbol{A}_k \boldsymbol{g} \\ s.t. \quad & \boldsymbol{g}^\tau \boldsymbol{g} \leqslant e_k^2 \,. \end{aligned} \tag{19}$$

Here $e_k > 0$ is a trust radius at the kth iteration, and

$$\boldsymbol{b}_k = -\sum_{i=1}^{M} \boldsymbol{L}_i[f^k](\boldsymbol{P}_i - \boldsymbol{X}_i^k) \,, \tag{20}$$

$$\boldsymbol{A}_k = \sum_{i=1}^{M} \boldsymbol{L}_i[f^k]\boldsymbol{L}_i[f^k]^\tau + w\boldsymbol{H} \,. \tag{21}$$

From the principles of the optimal condition [14], the trust region subproblem Eq. 19 is equivalent to the linear system of equations:

$$(\boldsymbol{A}_k + \mu_k \boldsymbol{I})\boldsymbol{g} = -\boldsymbol{b}_k \,, \tag{22}$$

with some $\mu_k \geqslant 0$ and $\boldsymbol{A}_k + \mu_k \boldsymbol{I}$ a positive definite matrix. One can easily solve $\boldsymbol{g}^k$ from the above equations.

Let D_k be the "predicted change":

$$D_k = Q^{(k)}(\boldsymbol{0}) - Q^{(k)}(\boldsymbol{g}^k) = \mu_k \|\boldsymbol{g}^k\|^2 + \frac{1}{2}\boldsymbol{g}^{k^\tau} \boldsymbol{A}_k \boldsymbol{g}^k \,. \tag{23}$$

Then we calculate the ratio

$$\rho_k = \frac{R(\boldsymbol{f}^k) - R(\boldsymbol{f}^k + \boldsymbol{g}^k)}{D_k} \tag{24}$$

of the actual change to the predicted change in the objective function $R(\boldsymbol{f})$. We set $\boldsymbol{f}^{k+1} = \boldsymbol{f}^k + \boldsymbol{g}^k$ if $\rho_k > 0$; otherwise $\boldsymbol{f}^{k+1} = \boldsymbol{f}^k$. It means that the trial step is acceptable whenever the objective function is reduced. Then a vector $\boldsymbol{b}_{k+1}$ and a symmetric matrix $\boldsymbol{A}_{k+1}$ are defined before finishing the kth iteration.

The choice of the next trust radius e_{k+1} depends on e_k and ρ_k. From Eq. 22, we can give a mode to select the trial step by changing parameter μ_k adaptively, which is equivalent to choosing e_k.

A formal description of our algorithm for the curve reconstruction based on the trust region technique is as follows:

1. Initialize the coefficient vector $\boldsymbol{f}^0 \in \mathbb{R}^N$, and compute $\boldsymbol{b}_0$ and $\boldsymbol{A}_0$. Set $\mu_0 > \underline{\mu} > 0$ and $0 < \beta_1 < \beta_2 < 1$, and choose a threshold $\varepsilon > 0$. Let $k := 0$.
2. If $\|\boldsymbol{b}_k\| \leqslant \varepsilon$, terminate the algorithm. Otherwise solve the linear system of equations Eq. 22 to obtain $\boldsymbol{g}^k$.
3. Calculate the ratio ρ_k by Eq. 24 and set

$$\boldsymbol{f}^{k+1} = \begin{cases} \boldsymbol{f}^k + \boldsymbol{g}^k, & \text{if } \rho_k > 0, \\ \boldsymbol{f}^k, & \text{otherwise} \,. \end{cases}$$

and

$$\mu_{k+1} = \begin{cases} 4\mu_k, & \text{if } \rho_k < \beta_1, \\ \max(\frac{1}{2}\mu_k, \underline{\mu}), & \text{if } \rho_k > \beta_2, \\ \mu_k, & \text{otherwise} \,. \end{cases}$$

3. Generate $\boldsymbol{b}_{k+1}$, $\boldsymbol{A}_{k+1}$ according to Eq. 20 and Eq. 21. Set $k := k + 1$ and go to Step 2.

The constant μ_0, $\underline{\mu}$ and $\beta_i (i = 1, 2)$ can be chosen by users. The typical values are $\mu_0 = 1$, $\underline{\mu} = 10^{-5}$ and $\beta_1 = 0.25$, $\beta_2 = 0.75$.

According to the convergence analysis of the trust region algorithms in [20], the above algorithm could reach a local optimal solution corresponding to the minimization problem, and the resulting curve may have a large geometric error from the data set. In this case, knot insertion algorithm has to be adaptively applied in the reconstruction process. We will explore this issue in a forthcoming paper.

5 Implementation and examples

In the section, we discuss briefly the initial shape specification of the active model and illustrate our new algorithm for the implicit curve reconstruction with some examples.

5.1 Initialization

Before we implement our algorithm, we first need to set the knot vectors for the B-spline basis functions. We consider a slightly enlarged rectangle, which encloses the given point clouds. For simplicity, the knots are chosen uniformly along x and y axes in the rectangle.

As the first step of our algorithm for implicit curve reconstruction, an initial curve $V(f^0)$ is required. We present a specification, with little price, of the initial implicit curve by setting the control coefficients to be

$$f_j^0 = f_{(r-1)n+s}^0 = c_{rs} = \frac{r(m-r+1)}{m}\frac{s(n-s+1)}{n} - c_0 \,, \tag{25}$$

where $N = mn$, $j = 1, \ldots, N$, and the constant c_0 is properly chosen such that the initial active curve $V(f^0)$ encloses the target shape (as shown in Fig. 3).

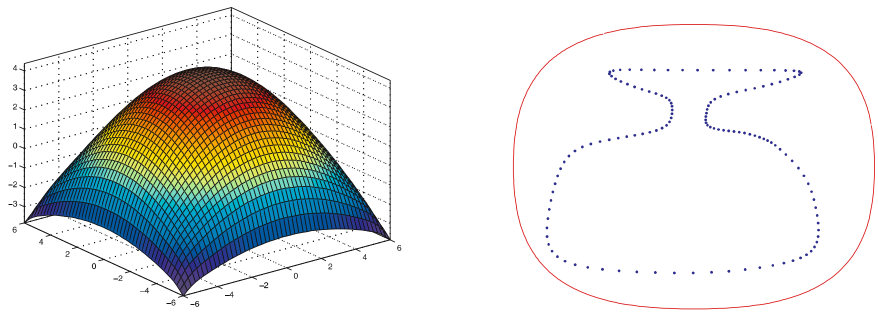
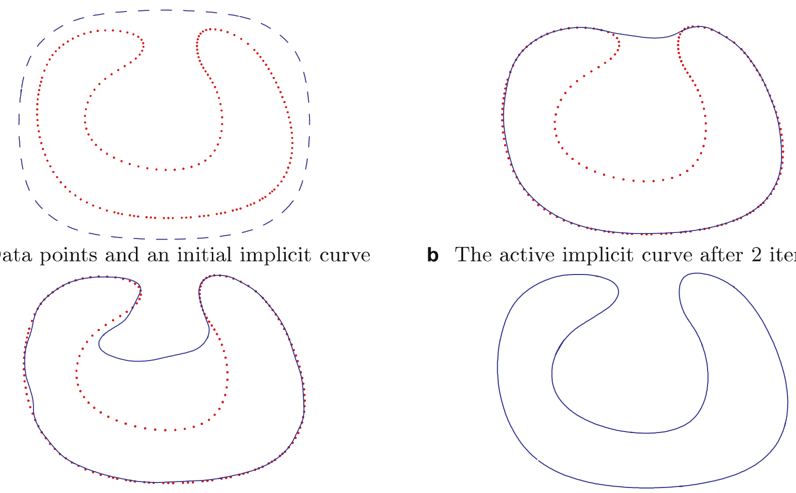
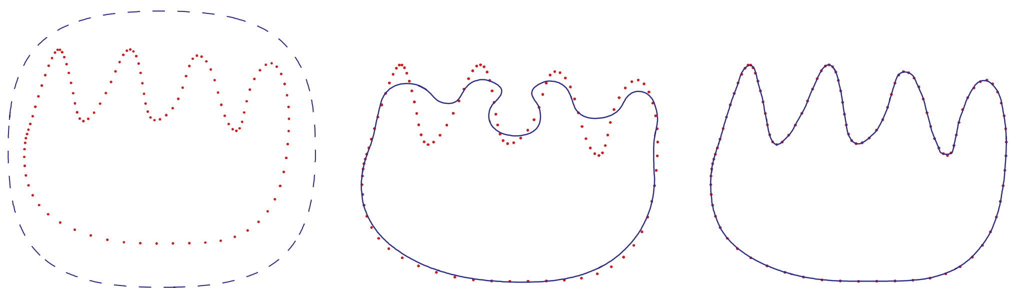


Fig. 3. The graph of $f^0(x, y)$ and the initial curve $V(f^0)$ enclosing the target shape



a Data points and an initial implicit curve
b The active implicit curve after 2 iterations
c The active implicit curve after 4 iterations
d The result implicit curve after 8 iterations

Fig. 4. Example 1: A concave shape



a Data and initial shape
b The immediate curve
c The final curve

Fig. 5. Example 2: Another concave shape

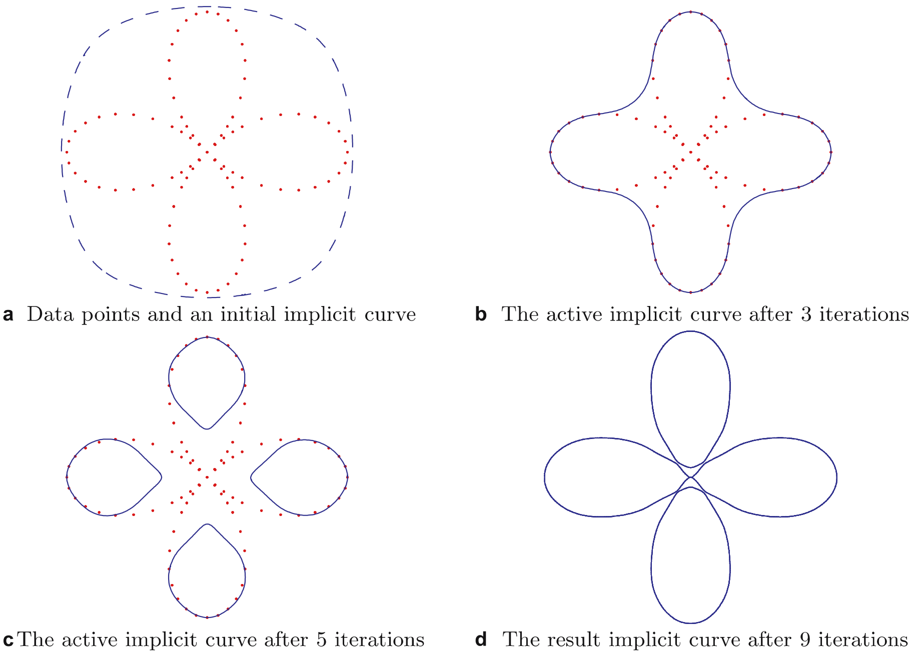


Fig. 6. Example 3: Rose curve

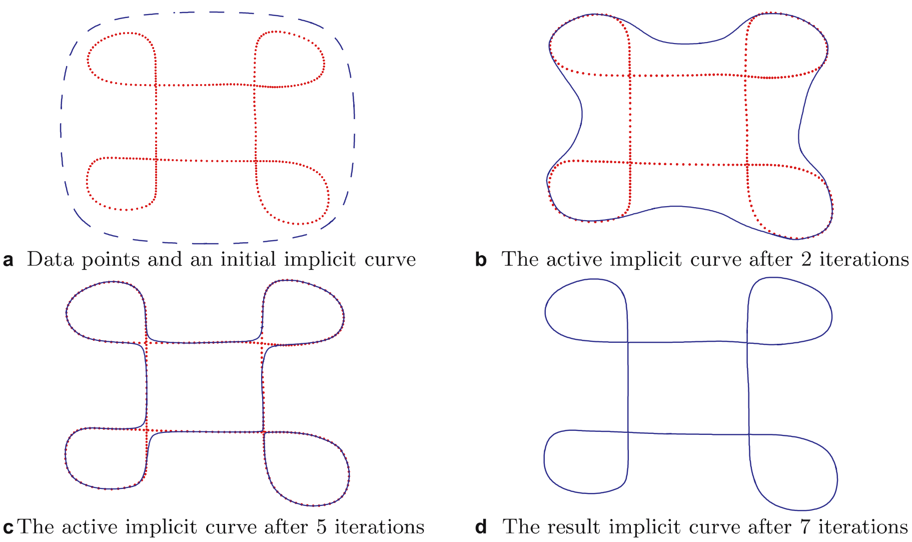


Fig. 7. Example 4: Chinese knot

In some situations, the convergence ratio depends heavily on the initial shape. A deeper analysis about the initial shape specification for the dynamic implicit curve reconstruction is presented in another paper [23].

5.2 Examples

In this subsection, we implement several examples to illustrate the effect of our implicit curve reconstruction technique. The results are presented in Figs. 4, 5, 6, 7, 8, and 9. In all examples, we display the data points and the initial shape followed by one or two intermediate shapes and the final shapes. The examples demonstrate several advantages of our implicit curve reconstruction algorithm. First, it can easily adapt to data sets with complicate topology and geometry. Second, it can handle data sets with noise. Finally, the algorithm converges in the most common circumstances.

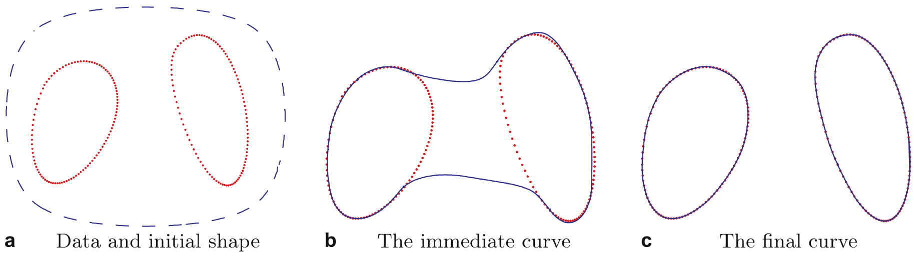


Fig. 8. Example 5: Disconnected target

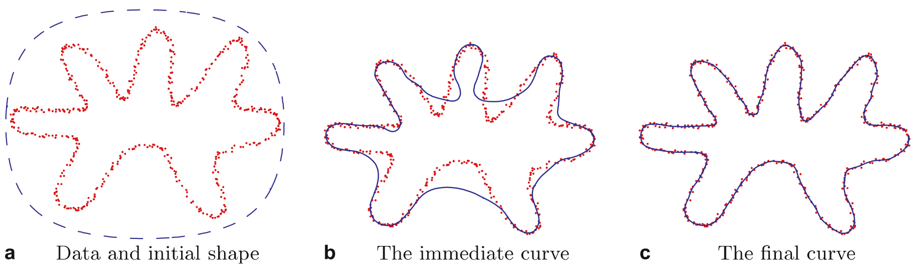


Fig. 9. Example 6: Data with noise

6 Conclusions and future works

In this paper, an active model for implicit curve reconstruction is proposed. The implicit curve is represented as the zero set of a bivariate tensor-product B-spline function, and the active model is based on a local approximation of the geometric distance error function. By minimizing the geometric distance between the point set and the implicit curve and some energy term, a sequence of curves are generated to approximate the target shape. The trust region algorithm from optimization theory is applied to solve the minimization problem. We test many examples, which show several advantages of our algorithm, such as, the ability to cope with data sets with complex topology and geometry, insensible to noises and robustness in most circumstances.

There are still some problems that need further investigation in the proposed curve reconstruction model.

– Currently, the knot sequences along x and y directions are chosen to be distributed uniformly. In the future, we will allow, during the iteration process, knot insertions to produce adaptive curve reconstruction.
– Since we have to compute the foot-point for each point in the point clouds and a non-linear optimization problem has to be solved, the proposed algorithm is a little time-consuming. It is worthwhile to improve the efficiency of the algorithm.
– The current model does not treat the singular points of the reconstruction curve especially. This may result in incorrect topology structure near the singular points (see Fig. 6 for an example). A mechanism should be introduced to force the implicit curve reflect a reasonable structure around the singular points.
– We will also try to extend the current model to implicit surface reconstruction. Furthermore, it is worthwhile exploring the applications of the implicit tensor-product B-spline curves and surfaces in geometric modeling and computer graphics.

Acknowledgement The work is support by the Outstanding Youth Grant of NSF of China (No. 60225002), a National Key Basic Research Project of China (2004CB318000), NSF of China (10201030 and 60473132), the TRAPOYT in Higher Education Institute of MOE of China, and SRF for ROCS, SEM.

References

1. Bajaj, C.L., Bernardini, F., Xu, G.: Automatic reconstruction of surfaces and scalar fields from 3D scans. In: Proceedings of SIGGRAPH'95, pp. 109–118 (1995)
2. Blake, A., Isard, M.: Active Contours. Springer, Berlin Heidelberg New York (1998)
3. Bloomenthal, J.: Introduction to Implicit Surfaces. Morgan Kaufmann, San Francisco (1998)
4. Boissonnat, J.D., Cazals, F.: Smooth shape reconstruction via neighbor interpolation of distance functions. In: ACM Symposium on Computational Geometry, pp. 223–232 (2000)
5. Brunnet, G., Hagen, H., Santarelli, P.: Variational design of curves and surfaces. Surveys Math. Ind. **3**, 1–27 (1993)
6. Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum,

B.C., Evans, T.R.: Reconstruction and representation of 3D objects with radial basis functions. In: Proceedings of SIGGRAPH 01, pp. 6776 (2001)
7. Curless, B., Levoy, M.: A voltric method for building complex models from range images. In: Proceedings of SIGGRAPH'96, pp. 303–312 (1996)
8. Hartmann, E.: Numerical implicitization for intersection and G^n-continuous blending of surfaces. Comput. Aided Geom. Des. **15**, 377–397 (1998)
9. Hilton, A., Stoddart, A.J., Illingworth, J., Windeatt, T.: Implicit surface-based geometric fusion. Comput. Vis. Image Under. **69**, 273–291 (1998)
10. Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W.: Surface reconstruction from unorganized points. In: Proceedings of SIGGRAPH'92, pp. 71–78 (1992)
11. Hoschek, J., Lasser, D.: Fundamentals of Computer Aided Geometric Design. AK Peters, Wellesley, MA (1993)
12. Jüttler, B., Felis, A.: Least-squares fitting of algebraic spline surfaces. Adv. Comput. Math. **17**, 135–152 (2002)
13. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: active contour models. Int. J. Comput. Vis. **1**(4), 321–331 (1988)
14. Kuhn, H.W., Tucker, A.W.: Non-linear programming. In: Neyman, J. (ed.) Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, pp. 481492. University of California Press, Berkeley, CA (1951)
15. Ma, W.Y., Ruth, J.P.: Parameterization of randomly measured points for least squares fitting of B-spline curves and surfaces. Comput. Aided Geom. Des. **27**, 663–675 (1995)
16. Morse, B.S., Yoo, T.S., Chen, D.T., Rheingans, P., Subramanian, K.R.: Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions. In: SMI'01 Proceedings of the International Conference on Shape Modeling and Applications, pp. 8998. IEEE Computer Society, Washington, DC (2001)
17. Muraki, S.: Volumetric shape description of range data using blobby model. Comput. Graph. (Proc. SIGGRAPH) **25**, 227–235 (1991)
18. Pottmann, H., Hofer, M.: Geometry of the squared distance function to curves and surfaces. In: Hege, H,, Polthier, K. (eds.) Visualization and Mathematics III. pp. 223244. Springer, Berlin Heidelberg New York (2003)
19. Pottmann, H., Leopoldseder, S., Hofer, M.: Approximation with active B-spline curves and surfaces. In: Proceedings of Pacific Graphics, pp. 8–25 (2002)
20. Powell, M.J.D.: On the global convergence of trust region algorithms for unconstrained optimization. Math. Prog. **29**, 297–303 (1984)
21. Turk, G., O'Brien, J.: Shape transformation using variational implicit functions. In: SIGGRAPH '99, pp. 335342 (1999)
22. Wang, W., Pottmann, H., Liu, Y.: Fitting B-spline curves to point clouds by square distance minimization. ACM Trans. Graph., in press
23. Yang, Z.W., Wu, C.L., Deng, J.S., Chen, F.L.: Specification of initial shapes for dynamic implicit curve/surface reconstruction. In: Proceedings of 1st Korea-China Joint Conference on Geometric and Visual Computing, in press
24. Zhao, H.K., Osher, S., Merriman, B., Kang, M.: Implict and nonparametric shape reconstruction from unorganized data using a variational level set method. Comput. Vis. Image Under. **80**, 295–314 (2000)
25. Zhao, H.K., Osher, S.: Visualization, analysis and shape reconstruction of unorganized data sets. In: Osher, S., Paragios, N. (eds.) Geometric Level Set Methods in Imaging, Vision, and Graphics. Springer, Berlin Heidelberg New York (2002)

Z. YANG was born in Fujian, China, in 1975. He received his Ph.D. degree from the University of Science and Technology of China in 2005. His research interests include Computer Graphics and Optimization Theory.

J. DENG was born in Shangdong, China, in 1971. He received his Ph.D. degree from the University of Science and Technology of China in 1998. His research interests include Computer Aided Geometric Design and Computer Graphics.

F. CHEN was born in Anhui, China, in 1966. He received his Ph.D. degree from the University of Science and Technology of China in 1994. He is a professor in the Department of Mathematics at the University of Science and Technology of China. His research interests include Computer Aided Geometric Design and Computer Graphics.