

Topologically Faithful Fitting of Simple Closed Curves

Daniel Keren

Abstract—Implicit representations of curves have certain advantages over explicit representation, one of them being the ability to determine with ease whether a point is inside or outside the curve (*inside-outside functions*). However, save for some special cases, it is not known how to construct implicit representations which are guaranteed to preserve the curve's topology. As a result, points may be erroneously classified with respect to the curve. The paper offers to overcome this problem by using a representation which is guaranteed to yield the correct topology of a simple closed curve by using homeomorphic mappings of the plane to itself. If such a map carries the curve onto the unit circle, then a point is inside the curve if and only if its image is inside the unit circle.

Index Terms—Implicit fitting, topologically faithful fitting, Jordan-Schoenflies theorem.

1 INTRODUCTION

THE problem of representing free-form shapes as zero-sets of implicit functions has been studied in the realms of computer vision, graphics, and robotics. Implicit fitting has the advantage that it allows the user to very quickly determine whether a point is inside the shape or not, and also to approximate its distance from the shape. This is beneficial for obstacle avoidance, ray tracing, and object modeling. Implicit models can also be applied for object recognition by using invariants [16], [7], [4], [10], [5], [22]. In this work, the discussion will be restricted to free-form curves which are simple and closed; these cover a large class of interesting shapes.

Given a curve C , represented by discrete measurements $\{(x_k, y_k)\}_{k=1}^n$, the goal is to recover a function $f(x, y)$ such that its zero-set, defined by $Z_f = \{(x, y) / f(x, y) = 0\}$, approximates C . Usually, this is achieved by minimizing a cost function such as $\sum_{k=1}^n f^2(x_k, y_k)$. Better results are achieved by using cost functions which provide a more accurate approximation to the geometric distance between the zero-set and the points [9], [11], [20], [1], [17], [13], [15], [3], [8], [19].

A problem which has plagued implicit fitting is the following: Even if the zero-set approximates the data well, it may have a very different topology. The source of this problem lies in the fact that the cost functions used for fitting only measure whether the data points are close to the zero-set, but they do not relay anything about the zero-set's topology. For example, spurious components of the zero-set which lie far from the data are not penalized, neither are self-intersections, loops, etc. Various efforts were undertaken to solve this problem including a heuristic which searches for extraneous components and penalizes them [17], methods which seek to approximate not only the data, but also its gradients [6], [2], [12], [18], and restricting the fitting to polynomials which are guaranteed to have a "nice" zero-set [9]. However, all these methods are either restricted in the type of curves they can approximate, or are liable to fail and result in zero-sets with a different topology than that of the data they attempt to fit. Usually, it is desired to fit a simple closed curve. However, the fit may have a very different topology. For example, see [2], [9].

This paper offers an implicit fitting paradigm which solves this problem, as it is *guaranteed* to return a simple closed curve. The fit is actually "semexplicit;" a curve C is assigned a homeomorphism h from the plane \mathcal{R}^2 to itself, which maps C

onto the unit circle S^1 (recall that a function is a homeomorphism if it is 1-1, onto, continuous, and its inverse is continuous). Thus, to determine whether a point p is in the interior of C , it is enough to test whether $h(p)$ is in the interior of S^1 . If h is easy to compute, this provides a simple solution to the inside-outside problem for C , because it is trivial to determine whether a point is in the interior of S^1 . For the sake of brevity, a slight abuse of terminology will be adopted in the rest of this paper: "In C (or S^1)" will mean "in the interior of C (or S^1)."

Since h is from \mathcal{R}^2 to itself, it has two components, $h_1(x, y), h_2(x, y) : \mathcal{R}^2 \rightarrow \mathcal{R}$. If h maps $C = \{(x_k, y_k)\}$ closely onto the unit circle, this means that for all k , $(h_1(x_k, y_k), h_2(x_k, y_k))$ is close to the unit circle, hence $h_1^2(x_k, y_k) + h_2^2(x_k, y_k) \approx 1$. So, the zero-set of the implicit function $h_1^2(x, y) + h_2^2(x, y) - 1$ is a good approximation to C . Depending on the application, one may either use h as an inside-outside function, or use $h_1^2 + h_2^2 - 1$ as an implicit representation of the curve. Polynomial representations have attracted considerable interest; if h_1, h_2 are polynomials, so is $h_1^2 + h_2^2 - 1$.

1.1 Structure of the Paper

In Section 2, the mathematical foundations for the fitting algorithm are laid, and polynomial mappings discussed. Section 3 describes the fitting algorithm and, in Section 4, two additional families of mappings are described. Some examples are provided in Section 5, and conclusions are offered in Section 6.

Note: due to lack of space, all proofs are omitted. Those interested in them are welcome to e-mail the author at dkeren@cs.haifa.ac.il.

2 TOPOLOGICAL CONSIDERATIONS

The *Jordan-Schoenflies Curve Theorem* [21] states that for any simple closed curve C in the plane, there is a homeomorphic mapping of the plane to itself which maps C onto the unit circle S^1 . Also, the restrictions of this homeomorphism define homeomorphisms between the inside and outside of C and the inside and outside, respectively, of the unit circle. As depicted in Fig. 1, such a mapping allows the user to reduce the computation of the inside-outside function for C to the trivial computation of the inside-outside function for S^1 : p is in C iff its image under the mapping is in S^1 .

The proof of the Jordan-Schoenflies Theorem does not offer a mechanism to construct the desired homeomorphism. The rest of this paper deals with the question of computing homeomorphic mappings from the plane to itself under which the image of a given curve C approximates the unit circle. The following lemma is helpful in proving that some of the mappings used in this paper are indeed homeomorphisms:

Lemma 1. Let $h(x, y) : \mathcal{R}^2 \rightarrow \mathcal{R}^2$ be a 1-1 continuous onto mapping such that

$$\lim_{\|(x,y)\| \rightarrow \infty} \|h(x,y)\| = \infty$$

(such conditions will in general be referred to as limit conditions). Then, h is a homeomorphism of the plane onto itself.

2.1 Monotonic Functions and Polynomial Mappings

The following lemma shows how to construct homeomorphic mappings of the plane to itself from monotonic functions in one variable.

Theorem 1. Let $f_1(t), g_1(t), f_2(t), g_2(t)$ be continuous strictly monotonic increasing functions of one variable t , such that

$$\lim_{t \rightarrow -\infty} f_1(t) = \lim_{t \rightarrow \infty} f_2(t) = \lim_{t \rightarrow -\infty} g_1(t) = \lim_{t \rightarrow \infty} g_2(t) = \infty \quad (1)$$

$$\lim_{t \rightarrow -\infty} f_1(t) = \lim_{t \rightarrow \infty} f_2(t) = \lim_{t \rightarrow -\infty} g_1(t) = \lim_{t \rightarrow \infty} g_2(t) = -\infty \quad (2)$$

(note that conditions (1) and (2), together with the monotonicity and continuity conditions, imply that f_1, f_2, g_1, g_2 are 1-1 and onto from \mathcal{R} to itself).

• The author is with the Department of Computer Science, University of Haifa, Haifa 31905, Israel. E-mail: dkeren@cs.haifa.ac.il.

Manuscript received 5 Sept. 2001; revised 4 Apr. 2002; accepted 19 Aug. 2002. Recommended for acceptance by S. Sarkar.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 114904.

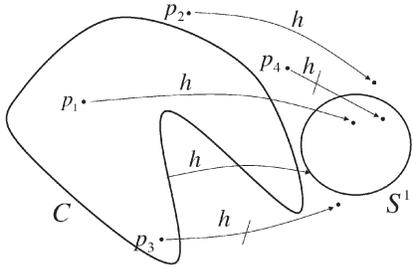


Fig. 1. If h is the homeomorphism guaranteed by the Jordan-Schoenflies Theorem, then one can determine whether a point p is in C by testing whether $h(p)$ is in S^1 (which is very easy to determine). For example, $p_1 \in C$ because $h(p_1) \in S^1$, and $p_2 \notin C$ because $h(p_2) \notin S^1$. Cases in which a point in C will be mapped to the exterior of S^1 (such as p_3), or vice-versa (such as p_4) are not possible.

Then, the following mapping from \mathcal{R}^2 to itself is 1-1 and onto:

$$(x, y) \rightarrow (f_1(x) + f_2(y), g_1(x) - g_2(y)). \quad (3)$$

The geometric idea behind the proof is outlined in Fig. 2.

Lemma 2. If $h(x, y) = (f_1(x) + f_2(y), g_1(x) - g_2(y))$, where f_1, f_2, g_1, g_2 are polynomial functions satisfying the conditions of Theorem 1, then $\lim_{\|(x,y)\| \rightarrow \infty} \|h(x, y)\| = \infty$.

Monotonic polynomials are natural candidates for the functions f_1, f_2, g_1, g_2 because they are easy to compute, they satisfy the limit conditions of Theorem 1 (hence, according to Lemmas 1 and 2, the mappings associated with them as defined in Lemma 2 are homeomorphisms), and because it is easy to characterize the polynomials that are strictly monotonically increasing—their derivatives must be everywhere positive. This condition can be enforced using the following observation.

Lemma 3. Let $p(x)$ be a polynomial in one variable which is everywhere positive. Then, there exist two polynomials $p_1(x), p_2(x)$ such that $p(x) = p_1^2(x) + p_2^2(x)$.

Combining these observations, it is easy to characterize everywhere, monotonically increasing polynomials of degree $2n+1$: Every such polynomial is the indefinite integral of $p^2(x) + q^2(x)$, where $p(x), q(x)$ are arbitrary n th degree polynomials. For example, every quintic monotonically increasing polynomial can be expressed as

$$\begin{aligned} c_5 x^5 + c_4 x^4 + c_3 x^3 + c_2 x^2 + c_1 x + c_0 &= \int [(a_2 x^2 + a_1 x + a_0)^2 + \\ &(b_2 x^2 + b_1 x + b_0)^2] dx = \left(\frac{1}{5} a_2^2 + \frac{1}{5} b_2^2\right) x^5 + \left(\frac{1}{2} a_1 a_2 + \frac{1}{2} b_1 b_2\right) x^4 + \\ &\left(\frac{2}{3} a_0 a_2 + \frac{1}{3} a_1^2 + \frac{2}{3} b_0 b_2 + \frac{1}{3} b_1^2\right) x^3 + (a_0 a_1 + b_0 b_1) x^2 + (a_0^2 + b_0^2) x + c, \end{aligned}$$

so every such polynomial can be parameterized by the following formulae

$$\begin{aligned} c_5 &= \frac{1}{5} a_2^2 + \frac{1}{5} b_2^2, \quad c_4 = \frac{1}{2} a_1 a_2 + \frac{1}{2} b_1 b_2, \\ c_3 &= \frac{2}{3} a_0 a_2 + \frac{1}{3} a_1^2 + \frac{2}{3} b_0 b_2 + \frac{1}{3} b_1^2, \\ c_2 &= a_0 a_1 + b_0 b_1, \quad c_1 = a_0^2 + b_0^2, \quad c = c_0, \end{aligned} \quad (4)$$

where the a_i are free parameters. The representation may not be 1-1, but that is not a problem, as the goal is to parameterize all monotonically increasing polynomials. Note that, obviously, only odd degree polynomials can be everywhere monotonic.

3 THE FITTING ALGORITHM

Given a family of homeomorphisms h from the plane to itself, indexed by a family of parameters par , the paradigm for fitting a curve is straightforward. If the curve is given as a set of points

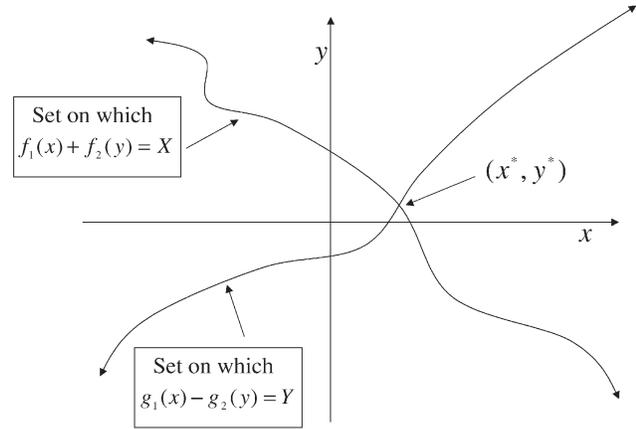


Fig. 2. A sketch of the geometric idea behind the proof of Theorem 1. The set of points (x, y) for which $f_1(x) + f_2(y) = X$ is a curve “extending from $(-\infty, \infty)$ to $(\infty, -\infty)$,” and the set of points (x, y) for which $g_1(x) - g_2(y) = Y$ is a curve “extending from $(-\infty, -\infty)$ to (∞, ∞) .” These two curves have to intersect, hence there’s a point (x^*, y^*) which satisfies both equations, so $h(x^*, y^*) = (X, Y)$.

$\{(x_k, y_k)\}_{k=1}^n$, then the following optimization problem should be solved

$$\arg \min_{par} \sum_{k=1}^n (\|h_{par}(x_k, y_k)\| - 1)^2, \quad (5)$$

if the minimum is close to zero, this means that C was mapped closely onto the unit circle, allowing for a representation as described in Section 2. The number of parameters varies according to the type of mapping h used. If, for example, a mapping like the one in (4) is used, then the total number of parameters is 28 (seven for each of the four quintics—the $a_2, a_1, a_0, b_2, b_1, b_0, c$ of (4)).

3.1 Stabilization

Experience has demonstrated that a straightforward minimization of (5) may yield a mediocre description for complicated shapes, even if the value of (5) is small; this happens because h may map a large portion of the curve’s points to a relatively small area near the circle. This means that h has very small derivatives in a large subset of the curve, which is not desirable for the following reason: p is determined to be inside C if $h(p)$ is inside S^1 . If h changes very slowly near C , the following situation may occur: $\|h(p)\| = 1 - \epsilon$ for a certain $p \in C$, for a small positive ϵ . But, if h changes slowly near p , then $h(p')$ may be smaller than 1 for points p' which are far from C ; and these points will also be determined to be in C , thus reducing the quality of the fit (see Fig. 3).

The method used to stabilize the fitting closely resembles the one used in [2], [6]. Regular behavior of the mapping near the curve was enforced by requiring that two “stripes” around it—internal and external—will be mapped to circles inside and outside the unit circle. To implement this, the expression in (5) is modified to

$$\begin{aligned} \arg \min_{par} \left[\sum_{k=1}^n (\|h_{par}(x_k, y_k)\| - 1)^2 + \sum_{l=1}^{n_1} (\|h_{par}(X_l, Y_l)\| - 1 + \epsilon)^2 + \right. \\ \left. \sum_{m=1}^{n_2} (\|h_{par}(X_m, Y_m)\| - 1 - \epsilon)^2 \right], \end{aligned} \quad (6)$$

where $\{(X_l, Y_l)\}$ and $\{(X_m, Y_m)\}$, respectively, range over an inner and outer stripe of C .

3.2 Optimization

In all tests, the Nelder-Mead method [14] was used to recover the optimal mapping. This holds true also for the mappings described in Section 4. The error term defined in (5) is nonpolynomial, due to the presence of the norm term. Replacing (5) (or, equivalently, all the summands in (6)) by $\arg \min_{par} \sum_{k=1}^n (\|h_{par}(x_k, y_k)\|^2 - 1)^2$,

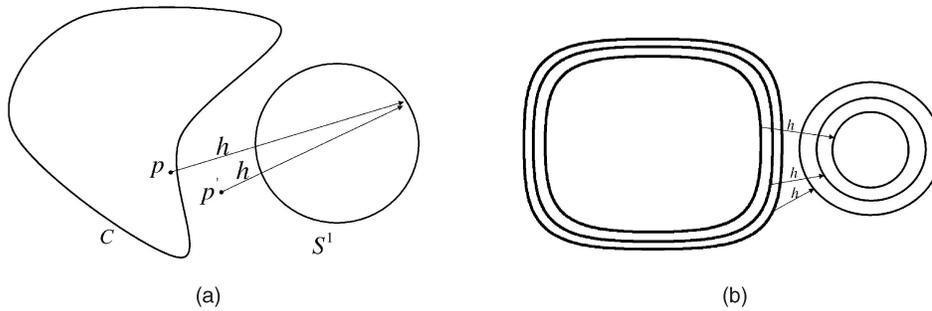


Fig. 3. (a) If h changes very slowly in the vicinity of the curve C , p' may be misclassified as lying inside it. (b) Stabilizing h by using “stripes.”

results in an eighth degree polynomial: The parameterization is quadratic in the free variables (see (4)), hence the norm squared is a quartic, and the squaring operation denoted by 2_* raises the total degree to eight. Therefore, many local minima may be present. The Nelder-Mead optimization method, combined with simulated annealing as presented in [14], is useful for “crawling out” of local minima.

3.3 Compositions of Homeomorphisms

Since the composition of two homeomorphisms is also a homeomorphism, complicated shapes may be fitted by composing a few mappings such as those described in Section 2. The drawback is that the degree of the composition of polynomial mappings is the product of the individual degrees; for example, composing three quintics results in a mapping of degree 125. Even though such mappings have far less degrees of freedom than a general polynomial mapping of degree 125, they tend to be unstable. A trade off should be sought between the number of mappings in the composite and their degrees, in order to avoid very high degrees in the composite. Eventually, this restriction limits the complexity of the shapes which can be described.

4 OTHER FAMILIES OF MAPPINGS

Two other families of mappings have been studied in addition to the polynomial mappings of Section 2. A short description follows.

4.1 Complex Polynomials

Since the curve can be viewed as lying inside the complex plane, it can be described by homeomorphisms of the complex plane to itself. It is clear, however, that there are no 1-1 polynomial maps from the complex plane to itself (save for the linear ones, which have no descriptive power). However, if the curve C is known to lie inside the unit disk D (which is trivial to obtain by scaling), it is easy to see that if h is 1-1 on D and, as before, h maps C to S^1 , then C can be described by $h^{-1}(S^1) \cap D$. A family of complex polynomials which are 1-1 on D , is provided by the following simple lemma:

Lemma 4. Let $P(z) = a_n z^n + a_{n-1} z^{n-1} + \dots + a_1 z + a_0$ be a polynomial with complex coefficients a_i . If

$$|a_1| > 2|a_2| + 3|a_3| + \dots + n|a_n|, \quad (7)$$

then $P(z)$ is 1-1 on the unit disk D .

It is straightforward to parameterize polynomials which satisfy (7). For example, once $a_2 \dots a_n$ were chosen, then a_1 can be parameterized by

$$a_1 = (2|a_2| + 3|a_3| + \dots + n|a_n| + \rho^2) \cos(\theta) + (2|a_2| + 3|a_3| + \dots + n|a_n| + \rho^2) \sin(\theta)i,$$

where θ and ρ are arbitrary real numbers.

4.2 Trigonometric Polynomials

For objects with a “wavy” boundary, which pose a formidable difficulty to polynomial representations, good results were obtained by using the same representation as in (3), but with f_1, f_2, g_1, g_2 trigonometric polynomials. As for algebraic polynomials, these trigonometric polynomials are guaranteed to be monotonically increasing by integrating a sum of squares of trigonometric polynomials. For example, a representation may be provided by

$$\begin{aligned} & \int [(a_1 \sin(x) + a_2 \sin(2x) + a_3 \sin(3x))^2 + (b_1 \sin(x) + b_2 \sin(2x) + \\ & b_3 \sin(3x))^2] dx = \\ & \frac{1}{2}(a_1^2 + a_2^2 + a_3^2 + b_1^2 + b_2^2 + b_3^2)x + (b_2 b_3 + b_1 b_2 + a_2 a_3 + a_1 a_2) \sin(x) + \\ & \frac{1}{2} \left(-\frac{1}{2} a_1^2 + a_1 a_3 + b_1 b_3 - \frac{1}{2} b_1^2 \right) \sin(2x) - \frac{1}{3} (b_1 b_2 + a_1 a_2) \sin(3x) - \\ & \frac{1}{4} \left(a_1 a_3 + \frac{1}{2} a_2^2 + \frac{1}{2} b_2^2 + b_1 b_3 \right) \sin(4x) - \frac{1}{5} (a_2 a_3 + b_2 b_3) \sin(5x) - \\ & \frac{1}{12} (a_3^2 + b_3^2) \sin(6x). \end{aligned}$$

Note that the limit conditions of Theorem 1 and Lemma 1 still hold because of the linear term, which surpasses the trigonometric terms as $x \rightarrow \infty$, just as the higher order term surpassed the other terms in the proof of Lemma 2. So, the proof of Lemma 2, as all the other previous results, still holds, and these trigonometric mappings can be used for faithful description.

5 RESULTS

The algorithm was tested on a few shapes, and its performance compared to that of the 3L fitting method proposed in [2]. The algorithm was also tested in the presence of noise.

5.1 The 3L Algorithm

The basic idea behind the 3L algorithm is to transform an implicit fitting problem to an explicit one. This is achieved by forcing the fitted polynomial to behave “nicely” at the vicinity of the shape, by making the shape’s distance transform approximate. Usually, two isodistance stripes are built around the shape—one on the inside and one on the outside (see also [6]). Then, an explicit fitting paradigm attempts to recover a polynomial which obtains a negative value on the inner stripe, a positive value on the outer stripe, and a value of zero on the shape itself. The great advantage of the 3L method is in its simplicity (the explicit fitting is a quadratic problem), and in that, it allows to achieve good control in the shape’s vicinity. However, it does not prevent the occurrence of extraneous components of the zero set.

5.2 The Shapes

Due to space limitations, the results for only four shapes are presented. Hopefully, they cover a spectrum of shape that is wide enough to evaluate the algorithm’s performance. There is one simple shape (“pear”), a more complicated shape (“hand”), a shape

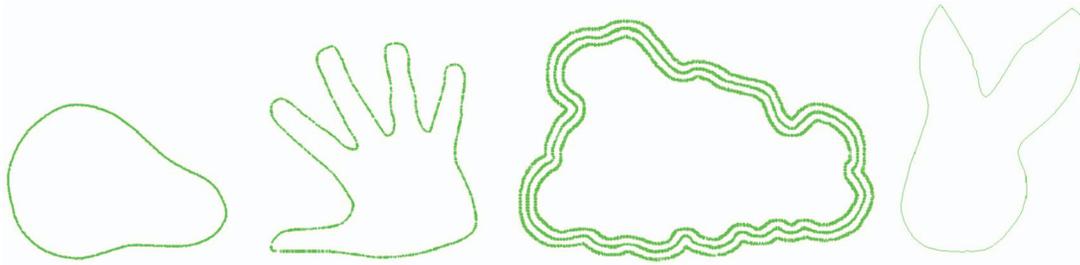


Fig. 4. Four shapes used in the fitting experiments: pear, hand, grape cluster, and bunny. The stripes used for computing the 3L fit to the grape cluster are also depicted.

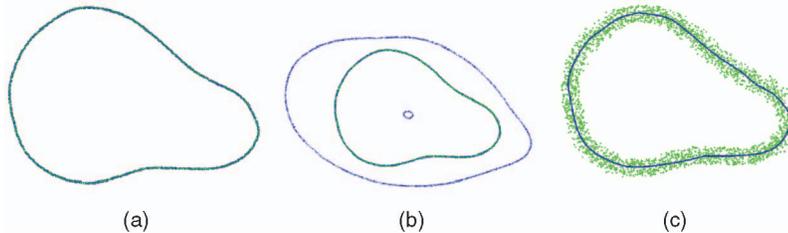


Fig. 5. (a) Pear shape (green) and topologically faithful fit (blue), using two complex polynomials of degree two (eight parameters). (b) 3L sixth degree fit (28 parameters). Note extraneous components. (c) Topologically faithful fit with strong Gaussian noise.

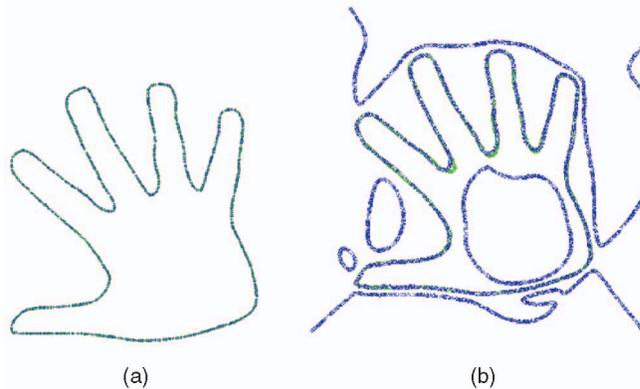


Fig. 6. (a) Hand shape (green) and topologically faithful fit (blue), using three polynomial mappings of degree five (60 parameters). (b) 3L 16th degree fit (153 parameters). Note extraneous components. Using polynomials with higher degrees did not improve the fit.

with a “wavy” boundary (“grape cluster”), and a shape with cusps (“bunny”). The shapes are depicted in Fig. 4. For each shape, the type of function used for the fitting, and the number of degrees of freedom (or parameters), are provided for the two methods. The topologically faithful fitting consistently required less degrees of freedom, hence resulting in a simpler, more compact description. For example, two complex polynomials of degree two were required for the “pear” shape (a total of eight parameters) when using topologically faithful fitting, while 3L fitting required a sixth-degree polynomial with 28 parameters. There were substantial differences in the other shapes as well. Possibly, this problem of 3L fitting could be alleviated by applying the observations in [16], to the effect that not all coefficients are important for the representation.

In all the examples, the data is depicted in green, and the fitted curve in blue. Since in many cases the fit overlaps with the data, usually the fitted curve is “thinned out.” This is achieved by randomly coloring half of the curve’s points with the color of the data. For better visual effect, the fit as well as the data are dilated to a width of four to five pixels. The results for the pear shape are depicted in Fig. 5, those for the hand shape in Fig. 6, and those for the grape cluster shape in Fig. 7.

5.3 Choosing the Model

The fitting algorithm described in this paper can be implemented for various types of functions described in Sections 2 and 4—polynomials, trigonometric functions, and complex polynomials. Moreover, each of these models has its own hierarchy (such as polynomial degree). The elaborate choice of the “correct” type and degree of

model, as undertaken in [16], is outside the scope of this paper. Roughly, the following rules of thumb can be offered: For simple shapes, complex polynomials of a low degree suffice; for objects with a “wavy” boundary, trigonometric polynomials achieve better results. As in the case of the “competing” 3L fitting, for every hierarchy of functions used, there is a saturation point beyond which the fit’s quality does not improve when the complexity of the model increases.

Note that the 3L fit falters at the “wavy” areas of the curve, and contains singularities. A certain stabilization of the 3L fit had to be applied. If the fitting is carried out by directly minimizing the target function associated with the explicit fitting described in Section 5.1, the result is very unstable, yielding a fit with many singularities in the zero set (Fig. 7b). In order to stabilize the solution, backward substitution with singular value decomposition (SVD) was used [14]. This solution results in a more stable zero set, however, its quality depends on the minimum allowed for the singular value in the SVD. When that value is too large, the fit’s quality decreases; in Fig. 8a, the result for a value of 10^{-6} is depicted. While the zero set is “nicer,” the fit moves away from the data. The result in Fig. 8b was the best I have been able to obtain, and it corresponds to a value of $4 \cdot 10^{-9}$.

In order to try and improve the 3L fitting result for objects with “wavy” boundaries, I have also implemented it with trigonometric functions—instead of a basis of the shape $x^n y^m$, a basis composed of the functions $\sin(nx) \cos(my)$, $\cos(nx) \sin(my)$, $\sin(nx) \sin(my)$, $\cos(nx) \cos(my)$ was used. This improved the results somewhat (Fig. 8c). However, the finer “wiggles” in the boundary are still missed. It is interesting to note that, for the grape cluster shape,

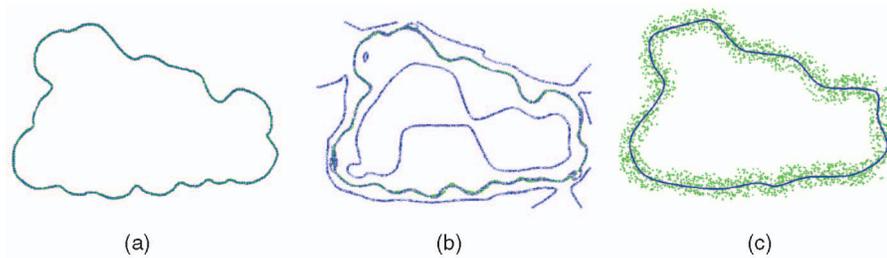


Fig. 7. (a) Grape cluster shape (green) and topologically faithful fit (blue), using two trigonometric mappings with a total of 128 parameters. (b) 3L 20th degree fit (231 parameters). Note extraneous components. Using polynomials with higher degrees did not improve the fit. (c) Topologically faithful fit with strong Gaussian noise.

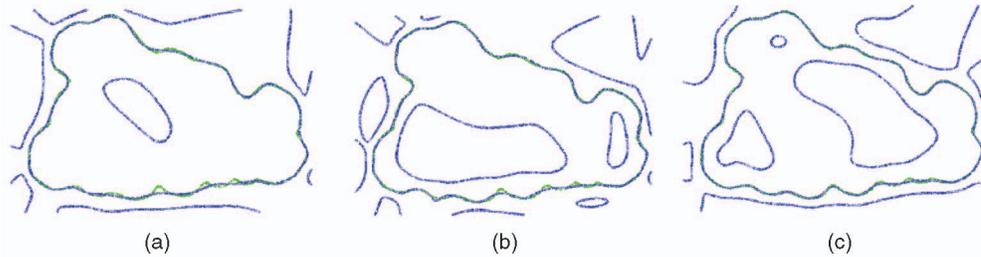


Fig. 8. (a) Grape cluster shape (green) and 3L 20th degree fit with a 10^{-6} lower limit on singular values (blue). (b) 3L 20th degree fit with a $4 \cdot 10^{-9}$ lower limit on singular values, which represents the best 3L polynomial fit obtained in the experiments. Note extraneous components. Using polynomials with higher degrees did not improve the fit. (c) 3L fit with trigonometric functions, 180 parameters.

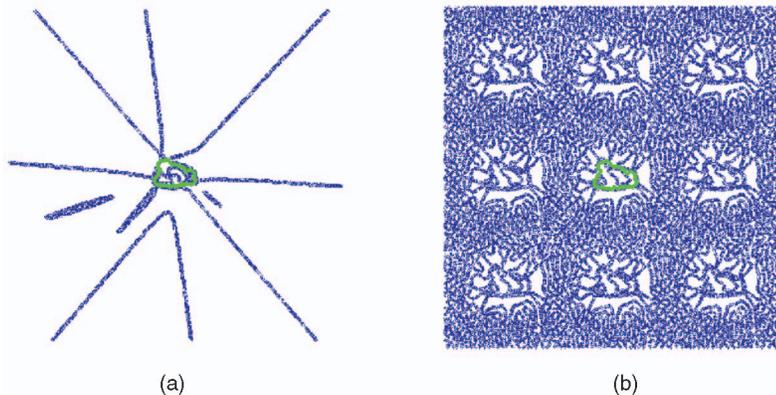


Fig. 9. Global behavior of 3L fit for grape cluster shape, with (a) a polynomial basis and (b) trigonometric.

trigonometric functions yield better results both for the 3L fitting and for the topologically faithful method.

5.4 Global Behavior of the 3L Fit

In order to depict the global behavior of the 3L fit, a plot of the zero sets for the best 3L polynomial and trigonometric fits to the grape cluster over a larger area are included (Fig. 9). As can be expected, the zero set of the trigonometric fit is periodic in nature. In both cases, the topology of the zero set is very different from that of the curve.

5.5 A Shape with Cusps

A result for a shape with cusps is provided next; this is the “bunny” shape (Fig. 10). Both algorithms “smooth out” the cusps at the ears’ ends, which are apparent in the original shape (Fig. 4). A different method of display was chosen here, since the difference between the data and the fits is only in the very fine details.

Although, theoretically, cusps can be fit by implicit polynomials (such as $x^2 + y^3$), in practice, they pose a difficult problem for implicit fitting, which tends to round or “smooth out” the cusps. This problem was also exhibited by the topologically faithful method suggested in this paper.

6 CONCLUSIONS AND FUTURE WORK

A novel method for fitting simple closed curves was presented. The method uses homeomorphic mappings of the plane to itself, which

map the curve close to the unit circle. These maps allow the quick computation of the curve’s inside-outside function, and also define an implicit representation of the curve. The main novelty and advantage is that the fit is guaranteed to have the same topology as the curve. The method was compared with the 3L fitting method, and while the latter is faster and simpler to implement, the

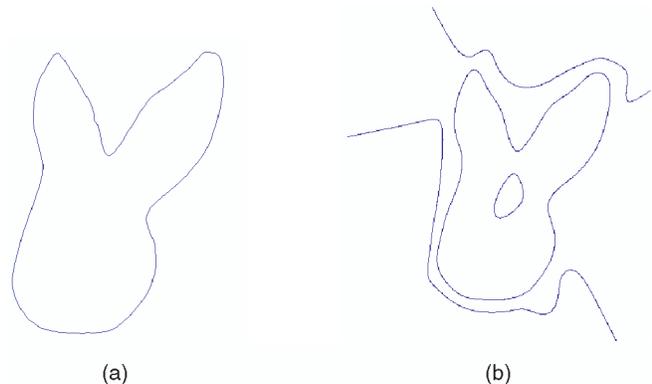


Fig. 10. (a) Topologically faithful fit for bunny shape, using two polynomial mappings of degree five (40 parameters). (b) Tenth degree 3L fit (66 parameters).

descriptions yielded by the suggested method are usually more compact than the 3L solution, they approximate the data somewhat better, and they do not suffer from extraneous components. Behavior under noise was "reasonable" in that the fit did not exhibit bias under noise. When the noise is very large and the fine outlines of the shape destroyed—as in the "grape cluster" shape in Fig. 7—the fitting finds a curve which passes in the "center" of the noisy points, thus approximating the rough outline of the original shape. The method is limited in the complexity of shapes it can describe, since composing many homeomorphisms results in very high degrees which cause numerical instabilities. Also, the fitting—like the 3L and other implicit fitting methods—tends to somewhat "smooth out" cusps.

Future work will first concentrate on extending these results to surfaces.

REFERENCES

- [1] C. Bajaj, I. Ihm, and J. Warren, "Higher-Order Interpolation and Least-Squares Approximation Using Implicit Algebraic Surfaces," *ACM Trans. Graphics*, vol. 12, no. 4, pp. 327-347, 1993.
- [2] M.M. Blane, Z. Lei, H. Civi, and D.B. Cooper, "The 3L Algorithm for Fitting Implicit Polynomial Curves and Surfaces to Data," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 3, pp. 298-313, 2000.
- [3] F.L. Bookstein, "Fitting Conic Sections to Scattered Data," *Computer Vision, Graphics, and Image Processing*, vol. 9, pp. 56-71, 1979.
- [4] D. Forsyth, J.L. Mundy, A. Zisserman, C. Coelho, A. Heller, and C. Rothwell, "Invariant Descriptors for 3D Object Recognition and Pose," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 10, pp. 971-992, Oct. 1991.
- [5] D.A. Forsyth, "Recognizing Algebraic Surfaces from Their Outlines," *Proc. Int'l Conf. Computer Vision*, pp. 476-480, May 1993.
- [6] C. Gotsman and D. Keren, "Tight Fitting of Convex Polyhedral Shapes," *Proc. Int'l J. Shape Modeling (special issue on Reverse Eng. Techniques)*, pp. 111-126, 1998.
- [7] D. Keren, "Using Symbolic Computation to Find Algebraic Invariants," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 11, pp. 1143-1149, Dec. 1994.
- [8] D. Keren, D.B. Cooper, and J. Subrahmonia, "Describing Complicated Objects by Implicit Polynomials," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 1, pp. 38-53, Jan. 1994.
- [9] D. Keren and C. Gotsman, "Fitting Curves and Surfaces with Constrained Implicit Polynomials," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 1, pp. 21-31, Jan. 1999.
- [10] D.J. Kriegman and J. Ponce, "On Recognizing and Positioning Curved 3D Objects from Image Contours," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 12, pp. 1127-1138, Dec. 1990.
- [11] Z. Lei and D.B. Cooper, "Linear Programming Fitting of Implicit Polynomials," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 2, pp. 212-217, Feb. 1998.
- [12] S. Muraki, "Volumetric Shape Description of Range Data Using Blobby Model," *ACM Computer Graphics (Proc. SIGGRAPH)*, vol. 25, no. 4, pp. 227-235, 1991.
- [13] V. Pratt, "Direct Least Squares Fitting of Algebraic Surfaces," *Computer Graphics*, vol. 21, pp. 145-152, July 1987.
- [14] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, *Numerical Recipes*. Cambridge Univ. Press, 1986.
- [15] P.D. Sampson, "Fitting Conic Sections to Very Scattered Data: An Iterative Improvement of the Bookstein Algorithm," *Computer Vision, Graphics, and Image Processing*, vol. 18, pp. 97-108, 1982.
- [16] J. Subrahmonia, D.B. Cooper, and D. Keren, "Practical Reliable Bayesian Recognition of 2D and 3D Objects Using Implicit Polynomials and Algebraic Invariants," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 5, pp. 505-519, May 1996.
- [17] S. Sullivan, L. Sandford, and J. Ponce, "Using Geometric Distance Fits for 3D Object Modeling and Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 12, pp. 1183-1196, Dec. 1994.
- [18] T. Tasdizen, J.-P. Tarel, and D.B. Cooper, "Improving the Stability of Algebraic Curves for Applications," *IEEE Trans. Image Processing*, vol. 9, no. 3, pp. 405-416, Mar. 2000.
- [19] G. Taubin, "Estimation of Planar Curves, Surfaces and Nonplanar Space Curves Defined by Implicit Equations, with Applications to Edge and Range Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 11, pp. 1115-1138, Nov. 1991.
- [20] G. Taubin, F. Cukierman, S. Sullivan, J. Ponce, and D.J. Kriegman, "Parameterized Families of Polynomials for Bounded Algebraic Curve and Surface Fitting," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 3, pp. 287-303, Mar. 1994.
- [21] C. Thomassen, "The Jordan-Schoenflies Theorem and the Classification of Surfaces," *Am. Math. Monthly*, vol. 99, no. 2, pp. 116-130, 1992.
- [22] M. Unel and W.A. Wolovich, "A New Representation for Quartic Curves and Complete Sets of Geometric Invariants," *Int'l J. Pattern Recognition and Artificial Intelligence*, vol. 13, no. 8, pp. 1137-1150, 1999.