TR-91-009

Approximation to Planar Curves by the Mininun Number of Piecewise Parabolas

Approximation to Planar Curves by the Minimum Number of Piecewise Parabolas

Mark C. K. Yang, Institute of Information Science, Academia Sinica, and
Department of Statistics, University of Florida, Gainesville, Fl 32611

Hsin-Min Sun, Institute of Information Science, Academia Sinica,
Taipei, Taiwan, ROC,

Abstract - Polygons and piecewise line segments are often used to represent planar curves. Though they are generally easy to compute, lack of slope continuity makes the curve look unsmooth. On the other hand, B-spline type of fitting can produce smooth curves, but their equations make optimality difficult to establish. In this paper, parabolas are used to approximate curves with continuous slopes, except at those points pre- or post- assigned as corners. An algorithm that provides the least number of parabolas is derived. Though it takes more computing time, the curves look more natural than the line fitting, for most objects.

Index Terms - Parabolic approximation, conic representation, digitized curve fitting.

I. INTRODUCTION

A typical digitized curve fitting problem can be described as follows:

Given a set of consecutive n points $S=\{ p_i = (x_i, y_i), i=1,2,..., n\}$ on the XY plane, the purpose is to find a curve C satisfying a discrepancy requirement $d(C, p_i) \leq \epsilon$ for all $i=1,..., n$, where ϵ is a given threshold and $d(C, p)$ denotes the distance between curve C and point p. (1.1)

Many methods have been proposed, such as using linear segments [1] - [4], B-splines [5] -

[8], and conic curves [9]- [10]. While the linear approximation is easy to compute, it lacks the smoothness required by most natural objects. On the other hand, the B-spline and conic approximations are smooth but the requirement $d(C, p_i) \leq \epsilon$ is difficult to justify. Consequently, the optimality of a fitting, such as the minimum number criterion, becomes very difficult to determine. In this paper, the dynamic programming method of Dunham[4] for line fitting is extended to piecewise parabolas. Except in some pre- or post-determined corner points, the fitted curve is continuous in slope and contains the minimum possible number of parabolas under certain constraints.

II. METHOD

Two terms have to be defined first.

Since $\tan\theta=\tan(\pi+\theta)$, mere slope continuity may produce $\succ$ type unsmooth connections. Thus, the slope in this paper exists in two forms; one is the direction vector $\vec{V}$, which distinguishes θ and $\pi+\theta$ and the other is the usual definition $m=\tan\theta$. The directional vector $\vec{V}$ at point p (the subscript i is omitted for a general description) is defined as follows: The left vector $\vec{V}_\ell$ is the unit vector pointing to p from the direction of the longest straight line, formed by p and its previous points, under the discrepancy requirement (see also [4], §III). Similarly, the right vector $\vec{V}_r$ is the unit vector defined by the longest straight line from p to the points after p satisfying the discrepancy requirement. Thus, $\vec{V}$ is defined as their average, i.e.,

$$\vec{V} = \frac{\vec{V}_r + \vec{V}_\ell}{|\vec{V}_r + \vec{V}_\ell|} \tag{2.1}$$

We also use the common unit vector notation (a, b), $a^2+b^2=1$, to represent $\vec{V}$. For points at the end, the average is replaced by a single angle.

Secondly, we define a feasible set of point p_k by

$$F_k = \{ p_{k+1}, p_{k+2}, \dots, p_{k+j} \},$$

where j is the largest integer such that it may still be possible to find a parabola to pass p_k with slope m_k and to satisfy the discrepancy criterion for all the points $p_{k+1}, \dots, p_{k+j}$.

The construction of F_k is given in § III.2.

We find that by a simple example, it is easier to describe the method. All the mathematical formulae and derivations are given in the next section. Let S containing n=12 points be an open curve, i.e., there is no connection between p_1 and p_n. With each point p_k, two indices are assigned; the number index ν_k and the terminal index t_k. The index number ν_k represents the smallest possible number of parabolas from p_k to p_n and t_k denotes the other terminal of the parabola that connects p_k and p_{t_k}. Let all $\nu_k=0$ and $t_k=0$ for all k=1,2, ..., n, before the algorithm starts (Table I (a)). The algorithm starts with p_{n-1}.

1) Change $\nu_{n-1}=1$ and $t_{n-1}=n$.

2) Repeat the fitting process from $p_{n-2}, p_{n-3}, \dots,$ to p_1 consecutively. At p_k, we fit it with a point in F_k according to the following order:

Higher priorities are given to the points with smaller number indices. For points with the same number index, higher priorities are given to the points closer to p_k. Table I (b) illustrates the fitting process at point p_5. Suppose the feasible set $F_5=\{6, 7, 8, 9\}$. Then the fitting priorities are 9, 7, 8, and 6.

Let the point now considered be $x \epsilon F_k$. With points x and p_k, and their corresponding slopes, a unique parabola can be found (§ III.1). The discrepancy requirement is checked to all the points between p_k and x (§ III.3). If the discrepancy requirement is satisfied, save the parabola equation, let $\nu_k=\nu_x+1$, $t_k= x$, and go to the point p_{k-1}. Otherwise, go to the point with the next priority. The process will stop because it is always possible to fit p_k and p_{k+1} with a parabola satisfying the slope and discrepancy requirement.

3) At the end , ν_1 denotes the minimum number of parabolas and t_1 can be used to trace all the parabolas from 1 to n. In Table I (c), we find that four parabolas are necessary to appoximate the set S. They are from 1 to 4, 4 to 7, 7 to 9, and 9 to 12.

Since at every stage k, the least number of parabolas is used to connect p_k and p_n, it is obvious that the answer at the last stage gives a minimum number parabola fit.

For a long curve, it is sometimes difficult to avoid corners. In other words, there are points where a smooth curve fitting is not as natural as a curve with sharp changes of slopes. While there is no clear-cut definition of a corner [11]-[13], the following definition is reasonable: Using the notation in (2.1), if

the angle between $\vec{V_r}$ and $\vec{V_\ell} \geq$ a threshold,

then we define p as a corner. Again, the value of the threshold is subjective. We feel $\pi/2$ is a reasonable choice. Once there is a corner point, fitting S is the same as fitting two open curves with one endpoint in common. Moreover, a closed curve with one corner point can be treated as a open curve. It is straightforward to extend the fitting method to more than one corner point.

If S is a closed curve without a corner point, finding a minimum set of parabolas is more time consuming. Let I_p be the set of all the points that contain p as their feasible point, i.e.,

$$I_p = \{ x \mid p \text{ belongs to } F_x \}.$$

Since one parabola has to pass p within the ϵ neighborhood, one of the starting points has to be in I_p. A minimum set of parabolas can be found exhaustively by trying all the points in I_p as starting points. To save computing time, the I_p with the minimum number of points should be chosen. It can be intuitively seen that a "corner" type of point is a strong candidate for an I_p with the minimum number of elements. For example, if p is nearly a corner, then I_{p+1} may contain the point p only. Thus, the exhaustive search in the parabola case does not usually contain as many points as the exhaustive search in the segment fitting ([4], § II, last paragraph).

III. FORMULAE AND DERIVATIONS

III.1 Fitting a parabola passing two points with given directions

Let the two points be $p_1=(x_1, y_1)$ and $p_2=(x_2, y_2)$, with unit directional vectors (a_1, b_1) and (a_2, b_2) respectively. Using the simple transformation of coordinates,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_1 & b_1 \\ -b_1 & a_1 \end{bmatrix} \begin{bmatrix} x-x_1 \\ y-y_1 \end{bmatrix},$$

we can transform point p_1 to the origin with direction $\overrightarrow{V_1}=(1, 0)$ and slope $m_1=0$. To simplify the notation, we eliminate the primes and still let $p_2=(x_2, y_2)$ with direction $\overrightarrow{V_2}=(a_2, b_2)$, and slope $m_2=b_2/a_2$. It can be easily shown that the equation for a parabola satisfying this condition for p_1 is of the form

$$(x + By)^2 + Ey=0. \tag{3.1}$$

Suppose $m_2 \neq 0$ or y_2/x_2. Then the unique parabola passing through p_1 and p_2 with corresponding slopes m_1 and m_2 is

$$B= \frac{x_2}{y_2} - \frac{2}{m_2},\ E= -2(x_2+By_2)(1+Bm_2)/m_2.$$

The derivation of B and E is straightforward. Moreover, the directional requirement is satisfied iff (if and only if)

(i) $x_2 + By_2 > 0$, and

(ii) $a_2 + Bb_2 > 0$. (3.2)

The proof of (3.2) seems nontrivial. We include a proof in VI.1.

If m_2 equals or nearly equals to y_2/x_2, the parabola becomes a straight or nearly a straight line. The continuity of slope at p_1 is marginal, but should be acceptable. However, if m_2 is close to 0 or (3.2) fails, then the parabola is very schewed. Fortunately, if this happens to two unadjacent points, the parabola should be discarded and the next candidate from the feasible set should be used. However, if p_2 is adjacent to p_1 and is the last resort in F_1, then we can simply connect p_1 and p_2 by a straight line. Actually, the only situation the latter occurs is when the left and the right lines from p_1 and p_2 have the same slope. Moreover, two adjacent points are usually too close to affect the smoothness.

III.2 Construction of the feasible set F_k

Using the same argument as III.1, we let the point $p_k=(0, 0)$ with slope $m_k=0$. Apparently, p_{k+1} is in F_k. We will add points to F_k by induction. Suppose $p_{k+1}, \ldots, p_{k+\ell}$ are known to be in F_k. We wish to know whether $p_{k+\ell+1}$ can be added to F_k. Ideally, one would like to check whether it is still possible to find B and E in (3.1) such that this parabola can pass through all the points $p_{k+1}, p_{k+2}, \ldots,$ and $p_{k+\ell+1}$ within the ϵ discrepancy. However, due to the mathematical difficulties in dealing with E, only B is examined. Thus, the feasible set may be larger than what is actually needed, but it is a valid one.

Let the feasible range for B, when the requirement is to pass only two points p_i

and p_j within the discrepancy ϵ, be B_{ij}. Then it is easy to see that if the set

$$\bigcap_{i=k+2}^{k+\ell+1} \bigcap_{j=k+1}^{i-1} B_{ij} \tag{3.3}$$

is not empty, we may be able to find a parabola passing through p_k, ..., $p_{k+\ell+1}$ within ϵ-neighborhoods. Thus, $p_{k+\ell+1}$ belongs to F_k and we should continue to check $p_{k+\ell+2}$, else $F_k = \{ p_{k+1}, \ldots, p_{k+\ell} \}$ and F_k contains only these points.

The range B_{ij} can be determined by the following procedure. Define SUBROUTINE

(B^*):

$$R=\sqrt{MY_i/my_j}, \quad r=\sqrt{my_i/MY_j};$$

$$u=\begin{cases} mx_i - R \cdot MX_j & \text{if } MX_j \geq 0 \\ mx_i - r \cdot MX_j & \text{otherwise,} \end{cases}$$

$$U=\begin{cases} MX_i - r \cdot mx_j & \text{if } mx_j > 0 \\ MX_i - R \cdot mx_j & \text{otherwise.} \end{cases}$$

$$v= \min(\sqrt{my_i \cdot my_j} - my_i, \sqrt{MY_i \cdot my_j} - MY_i)$$

$$V= \max\{\sqrt{MY_j \cdot my_i} - my_i, \sqrt{MY_j \cdot MY_i} - MY_i, \delta_1, \delta_2\}, \text{ where}$$

$$\delta_1 = MY_j/4 \ \text{ if } \sqrt{MY_j/4} \in [\ \sqrt{my_i}\ ,\ \sqrt{MY_i}\ ]$$

$$\delta_2 = MY_i \quad \text{ if } \sqrt{MY_i} \in [\ \sqrt{my_j/4}\ ,\ \sqrt{MY_j/4}\ ]\ .$$

The output B_{ij} is the interval of B determined by Table II. Note that if $my_j=0$, then $R=\infty$, and consequently, if $MX_j \geq 0$, then $u=-\infty$, and if $mx_j \leq 0$, $U=\infty$.

Before computing B_{ij} by (B^*), the parameters MX_i, MX_j, etc. are defined from the coordinates of $p_i=(x_i, y_i)$ and $p_j=(x_j, y_j)$:

$$LX_i = x_i + \epsilon,\ \ sx_i = x_i - \epsilon,\ \ LX_j = x_j + \epsilon,\ \ sx_j = x_j - \epsilon;$$

$$LY_i = y_i + \epsilon,\ \ sy_i = y_i - \epsilon,\ \ LY_j = y_j + \epsilon,\ \ sy_j = y_j - \epsilon;$$

If $LY_i > 0$ and $LY_j > 0$,

then $MX_i=LX_i$, $MX_j=LX_j$, $mx_i= sx_i$, $mx_j= sx_j$

$MY_i=LY_i$, $MY_j=LY_j$, $my_i=\max(0, sy_i)$, $my_j=\max(0,sy_j)$

GO TO (B^*), let the output be B_1;

Else $B_1 = \phi$. (3.4a)

If $sy_i < 0$ and $sy_j < 0$,

then $MX_i=-sx_i$, $MX_j=-sx_j$, $mx_i= -LX_i$, $mx_j= -LX_j$

$MY_i=-sy_i$, $MY_j=-sy_j$, $my_i=\max(0, -LY_i)$, $my_j=\max(0, -LY_j)$

GO TO (B^*), let the output be B_2;

Else $B_2 = \phi$. (3.4b)

The range $B_{ij} = B_1 \cup B_2$.

Also, if $B_1 \neq \phi$ and $B_2 \neq \phi$, then B_{ij} is one interval (instead of possibly two). This fact helps in checking the emptiness of (3.3).

To justify the above algorithm, recall that equation (3.1) passes through (0, 0) with slope 0. If it also passes through $p_i=(x_i, y_i)$ and $p_j=(x_j, y_j)$, then the parameter B becomes

$$B = \frac{x_i - \gamma x_j}{\gamma y_j - y_i},\ \gamma = \pm \sqrt{y_i/y_j}\,. \tag{3.5}$$

Note that γ is defined only when $y_i y_j > 0$. In case $y_i y_j \leq 0$, the construction of a parabola is not possible. The ϕ set outputs in (3.4a,b) are the consequences of this. The $\pm$ sign for γ indicates that there are two parabolas that satisfy the requirements. However, it will be shown in §VI.2 that the only valid γ for our curve fitting is the positive one.

Since there is a leeway of ϵ, the parabola does not have to pass through p_i or p_j. However, it must pass through the two ϵ-neighborhoods. Due to the technical difficulties in handling circular neighborhoods, two squares $(x_i \pm \epsilon,\ y_i \pm \epsilon)$ and $(x_j \pm \epsilon,\ y_j \pm \epsilon)$ are used. Since the squares are larger than the required ϵ-neighborhoods, passing through the squares are only necessary. Thus, the feasible set so constructed may be larger than the real one. Again, it is a valid one. It can be seen that the MX_i, MX_j etc. in the algorithm are based on the two squares and the subroutine B^* is based on the maximum and minimun range arguments of B in (3.5) with $\gamma>0$. For example, the formula for V comes from the complicated maximum minimun structure of the denominator, $\gamma y_j - y_i = y_j/4 - (\sqrt{y_i} - \sqrt{y_j/4})^2$.

III.3 Finding the minimum distance between a point and a parabola

The basic idea is to transform the coordinates so that the parabola is in the standard form $y=ax^2$. Let the point be (x_0, y_0). Then we need to minimize $(x-x_0)^2 + (y - y_0)^2$ subject to $y - ax^2=0$. A standard Lagrange multipliers method will lead to solving a cubic equation. The formulas are available in most algebra books. If there is a single real root, the minimum distance can be easily obtained by substitution and if there are three real roots, then three locally minimum distances can be found. The least of them is the minimum distance. In case one or more normals from the curve do not intersect with the particular section of the parabola chosen for fitting, the endpoints of the parabola are used to compute the minimum distance.

IV. EXPERIMENTAL RESULTS

Three objects; a glass(Fig. 1), an apple(Fig. 2), and a cartoon silhoutte Snoopy with a baseball bat(Fig. 3) are used to illustrate the minimum number parabola algorithm. The main purpose is to compare it with the minimum segment representation [4] in curve quality, computational and storage complexities.

At the very left of each figure is the digitized curves with randomly generated uniform errors of 2 pixel length. The middle figures are the minimum segment fitting with $\epsilon=2$, and the right ones are the minimum parabola fit also with $\epsilon=2$. It is reasonable to say that the parabola fit is more natural. Table III presents the comparison of the two methods in terms of the number of segments and computing time. Note that for the line segments, only two coordinates of the endpoints need to be stored for reconstruction, while the parabolas need an additional slope at each end point. Take the glass for example, it takes 2x20=40 real numbers to saved all the line segments, but 3x13=39 real numbers to save the 13 parabolas. Which one uses a smaller number of storage depends on the curve. The line segment representation has some advantages if there are a large number of turns, such as in the Snoopy, but the parabolas have advantages for smooth curves, such as the glass. The biggest disadvantage of the parabola fitting is the slowness in computation. The numbers in the time column are in seconds based on the Turbo Pascal program running on an IBM-AT with Intel 80287 math coprocessor. During the three curve fittings, we did not encounter any odd situations such as $y_2=0$, $m_2=0$ or y_2/x_2, or directional reversion between two adjacent points.

V. DISCUSSION AND CONCLUDING REMARKS

This paper has derived an algorithm that uses the minimum number of parabolas to represent digitized figures. When compared with the minimum line segment representation, the trade-off is between picture quality and computational time.

Here are some comments on the two constraints given to the parabolas used for fitting; (i) all the parabolas end at a data point, and (ii) the slopes at the parabola connections are subject to the slopes estimated by (2.1). Without constraint (i), an extra section irrelevant to the data may be produced just to make the connection smooth. An example is shown in Figure 4. The same situation can also occur in line approximation [4] where all lines are forced to end at data points. Constraint (ii) does not exist in line fitting, because slope continuity is not required. Since slope estimation is always sensitive to noise, we can not find a better way to estimate the slopes. There may be some space for future improvement in this respect.

VI. PROOFS

VI.1 Proof of (3.2)

The main idea of our proof is to add directional information to a curve following the order it should be drawn. One of the easiest way to do this is to represent the curve in a parametric form such as: (i) $(f(t), g(t))$, $t \epsilon R$. If we fix the drawing sequence from $t=-\infty$ to $t=+\infty$, then (ii) $(f(-t), g(-t))$ will produce the same curve drawn in the reverse order. The two derivatives $(f'(t), g'(t))$, and $(-f'(-t), -g'(-t))$ reveal the two possible directions at any given point (with the same slope). Thus, with given order of the point sequence one has to choose one of the two representations (i) or (ii).

To parametrize (3.1), we rotate the coordinate system by the angle $\theta \equiv \tan^{-1}B$, with a uniquely determined $|\theta|<\pi/2$. Then point p_2 becomes

$$\left(\frac{x_2+By_2}{\sqrt{1+B^2}}, \frac{y_2-Bx_2}{\sqrt{1+B^2}}\right), \tag{6.1}$$

the directional vectors $(1, 0)$ of p_1 and (a_2, b_2) of p_2 become

$$\text{(i)}\quad \left(\frac{1}{\sqrt{1+B^2}},\ \frac{-B}{\sqrt{1+B^2}}\right), \text{ and}$$

$$\text{(ii)}\quad \left(\frac{a_2+Bb_2}{\sqrt{1+B^2}},\ \frac{b_2-Ba_2}{\sqrt{1+B^2}}\right), \tag{6.2}$$

respectively, and equation (3.1) becomes

$$x^2 + Dx + Fy = 0, \text{ with}$$

$$D = E\sin\theta/(1+B^2), \text{ and } F = E\cos\theta/(1+B^2). \tag{6.3}$$

Let the drawing sequence for t be from $-\infty$ to $+\infty$. Then the two parametric representation of (6.3) are

$$\left(t,\ -\frac{t^2 + Dt}{F} \right), \tag{6.4a}$$

$$\left(-t,\ -\frac{t^2 - Dt}{F} \right). \tag{6.4b}$$

By (6.2) (i) and the derivatives of (6.4a, b), we know that (6.4a) is the representation with the right direction at p_1. Since p_2 is drawn after p_1, the x-coordinate of (6.1) has to be >0. This verifies (3.2) (i).

Note that in general, two directions (α_1, β_1) and (α_2, β_2) with the same slope are identical iff $\alpha_1\alpha_2>0$, or $\alpha_1=\alpha_2=0$, and $\beta_1\beta_2>0$. Thus, in order for the direction at p_2 to be consistent, we need the sign consistency of (6.2) (ii) and the derivative of (6.4a). This is the same as (3.2) (ii). (3.2) is proven.

VI.2 Proof of $\gamma>0$ in (3.5).

No matter which parabola is chosen in (3.5), the point (0, 0) cuts the parabola into two parts. The parabola constructed by the minus γ puts p_i and p_j on the two different sides of (0, 0), while the one by the positive γ puts the two points on the same side. Since the former violates the sequence in S, the only valid γ is the positive one.

To see this, note that the slope at $p=(x, y)$ on the parabola (3.1) is

$$m_p = \frac{-2}{B-x/y} \tag{6. 5}$$

and the slope of the major axis of the parabola is $m_L=-1/B$. The two parts separated by (0, 0) are

Part1={ p on (3.1) | m_p is between 0 and m_L}, and

Part2={ p on (3.1) but not in Part1}.

The proof of p_i and p_j being on the same part iff $\gamma>0$, depends on two similar cases; $B<0$ and $B>0$. First, suppose $B<0$. Then $m_L>0$ and p belong to Part 1 iff $0<-B< x/y$, i.e., p_i and p_j belong to the same part iff $(B+x_i/y_i)\ (B+x_j/y_j) > 0$, which is equivalent to

$$\frac{\gamma}{y_i y_j}\left(\frac{x_i y_j - x_j y_i}{y_i - \gamma y_j}\right)^2 > 0.$$

Since $y_i y_j>0$, we must have $\gamma>0$. Identical result can be proven for $B>0$.

REFERENCES

[1] T. Pavlidis and S. L. Horowitz, "Segmentation of plane curves," *IEEE Trans. Comput.*, vol C-23, pp. 860-870, Aug. 1974.

[2] J. Sklansky and V. Gonzalez, "Fast polygonal approximation of digitized curves," *Pattern Recognition,* vol. 12, pp. 327-331, 1980.

[3] R. Bellman, "On the approximation of curves by line segments using dynamic programming," *Commun. ACM,* vol. 4, p.284, 1961.

[4] J. G. Dunham, "Optimum uniform piecewise linear approximation of planar curves," *IEEE Trans. Pattern Anal. Mach. Intell.,* vol PAMI-8, pp. 67-75, 1986.

[5] D. A. Ballard and C. M. Brown, *Computer Vision,* Prentice -Hall, Englewood Cliffs, N.J., 1982.

[6] T. Pavlidis, *Algorithms for Grahpics and Image Processing,* Comput. Sci., Rockville, Md, 1982.

[7] O. Lozover and K. Preiss, "Automatic generation of cubic B-spline representation for a general digitized curve," *Eurographics* (J. L. Encarnacao, Ed.), pp. 119-126, 1981.

[8] M. C. K. Yang, C. Kim, K. Cheng, C. Yang, and S. S. Liu, "Automatic curve fitting with quadratic B-spline functions and its applications to computer-assisted animation," *Comput. Vision, Graphics, Image Proc.,* vol. 33, pp. 346-363, 1986.

[9] T. Pavlidis, "Curve fitting with conic splines," *ACM Trans. on Graphics,* vol. 1, pp. 1-31, Jan. 1983.

[10] G. Farin, "Curvature continuity and offsets for piecewise conics," *ACM Trans. on Graphics,* vol. 8, pp. 89-99, April 1989.

[11] L. S. Davis, "Understanding shape: angles and slides," *IEEE Trans. Computers.* vol C-26, pp. 236-242, March, 1977.

[12] K. Rangarajan, M. Shah, and D. Van Brakcle, "Optimal corner detection," *Comput. Vision, Graphics, Image Proc.*, vol 48, pp. 230-245, 1989.

[13] J. A. Noble, "Finding corners," *Image and Vision Computing,* vol 6, pp. 121-128, May, 1988.

Legends for the Figures.

Figure 1. Digitized curve, Minimum Number (MN) Segment Line Fitting, and MN parabola Fitting for a Glass (segments separated by colors)

Figure 2. Digitized curve, MN Segment Line Fitting, and MN parabola Fitting for an Apple

Figure 3. Digitized curve, MN Segment Line Fitting, and MN parabola Fitting for a Cartoon, Snoopy with a baseball bat

Figure 4. Example of the Flaw When Floating Curves (Endpoints of Each Curve Section May End Anywhere) Are Used in Digitized Curve Fitting

TABLE I. AN EXAMPLE FOR MINIMUM NUMBER OF PARABOLIC FITTING

(a) In the Beginning

k=	1	2	3	4	5	6	7	8	9	10	11	12
ν_k=	0	0	0	0	0	0	0	0	0	0	0	0
t_k=	0	0	0	0	0	0	0	0	0	0	0	0

(b) For point 5

k=	1	2	3	4	5	6	7	8	9	10	11	12
ν_k=	0	0	0	0	0	3	2	2	1	2	1	0
t_k=	0	0	0	0	0	8	9	9	12	11	12	0

(c) At the end of the process

k=	1	2	3	4	5	6	7	8	9	10	11	12
ν_k=	4	4	4	3	3	3	2	2	1	2	1	0
t_k=	4	4	6	7	7	8	9	9	12	11	12	0

TABLE II. THE INTERVAL RANGE OF B

	$u \geq 0$	$u < 0 < U$	$U \leq 0$
$v > 0$	$(\frac{u}{V}, \frac{U}{v})$	$(\frac{u}{v}, \frac{U}{v})$	$(\frac{u}{v}, \frac{U}{V})$
$v = 0$	$(\frac{u}{V}, \infty)$	$(-\infty, \infty)$	$(-\infty, \frac{U}{V})$
$v < 0 < V$	$(-\infty, \infty)$	$(-\infty, \infty)$	$(-\infty, \infty)$
$V = 0$	$(-\infty, \frac{u}{v})$	$(-\infty, \infty)$	$(\frac{U}{v}, \infty)$
$V < 0$	$(\frac{U}{V}, \frac{u}{v})$	$(\frac{U}{V}, \frac{u}{V})$	$(\frac{U}{v}, \frac{u}{V})$

TABLE III. COMPARISONS OF LINE AND PARABOLIC FITTINGS

Object	Size(corner)	Line Fit		Parabola Fit	
		Number	Time	Number	Time
Glass	194 (0)	20	4.99	13	444.5
Apple	146 (2)	21	2.58	14	207.7
Snoppy	266 (5)	49	3.73	44	210.36

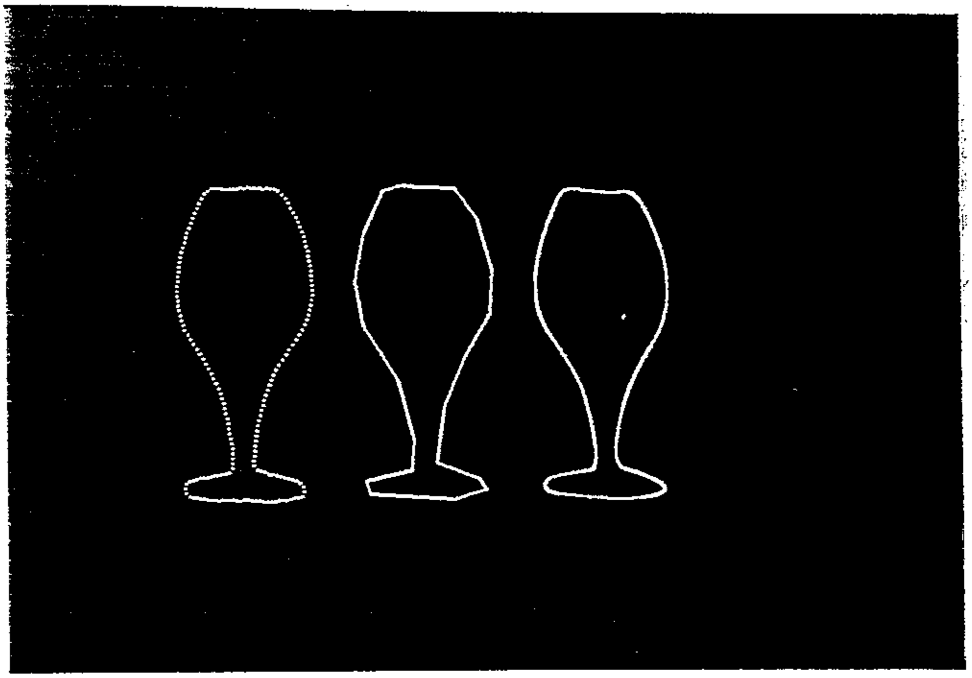

Fig. 1

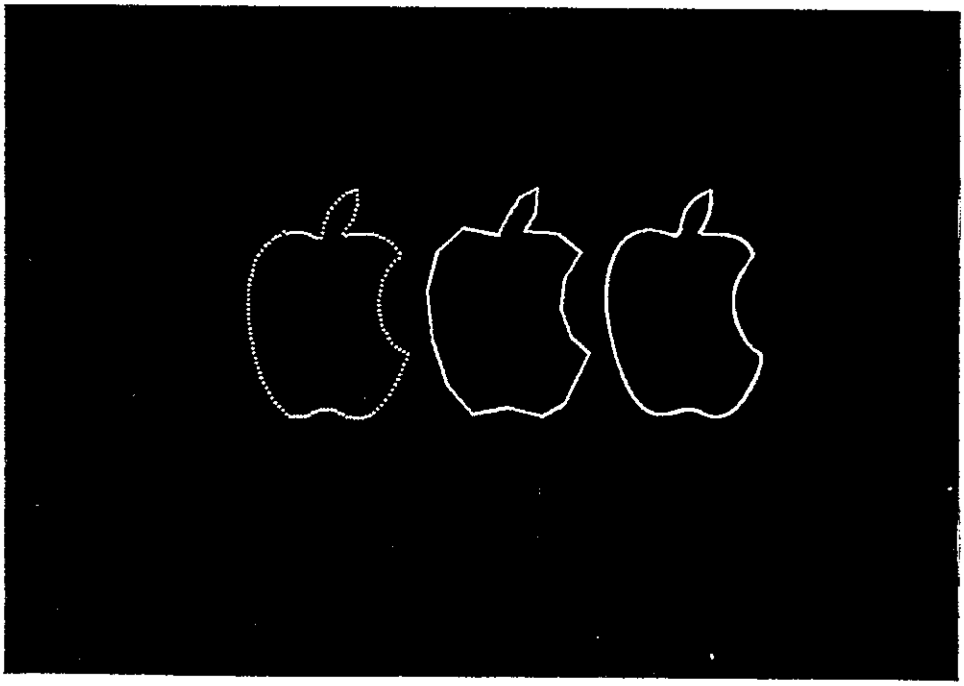

Fig. 2

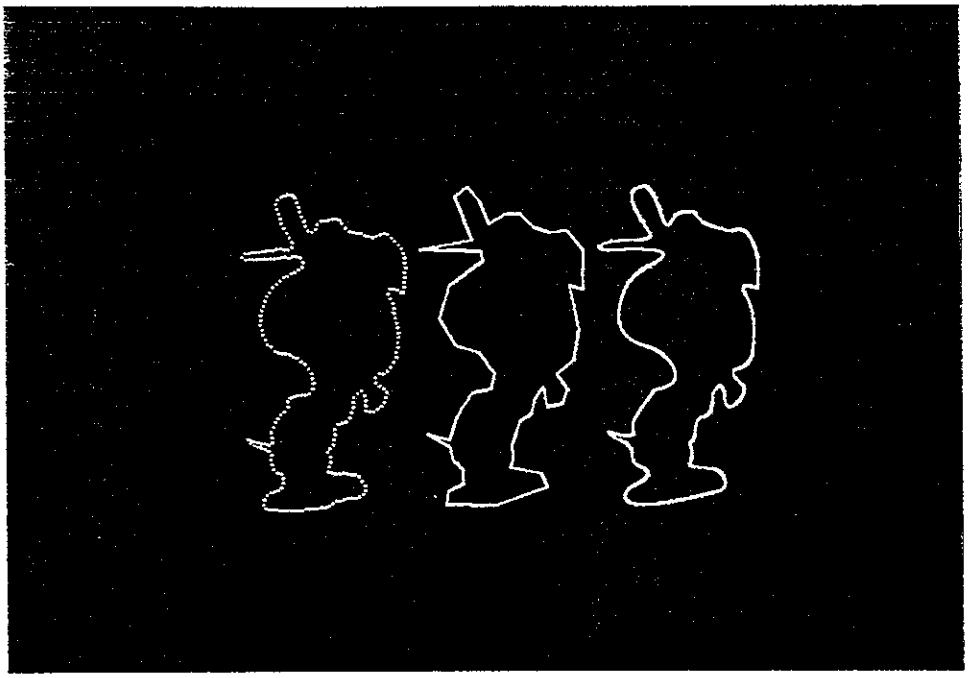

Fig. 3

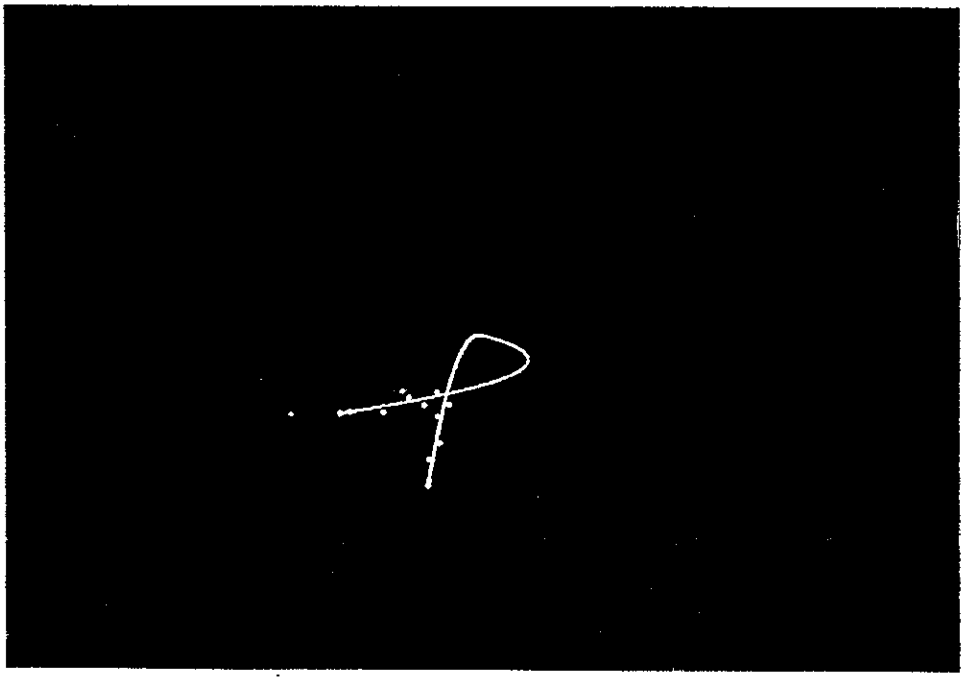

Fig. 4