Shape Discrimination Using Fourier Descriptors

ERIC PERSOON, MEMBER, IEEE, AND KING-SUN FU, FELLOW, IEEE

Abstract—**Description or discrimination of boundary curves (shapes) is an important problem in picture processing and pattern recognition Fourier descriptors (FD's) have interesting properties in this respect. First, a critical review is given of two kinds of FD's. Some properties of the FD's are given and a distance measure is proposed, in terms of FD's, that measures the difference between two boundary curves. It is shown how FD's can be used for obtaining skeletons of objects. Finally, experimental results are given in character recognition and machine parts recognition.**

I. Introduction

ONE OF THE PROBLEMS in picture processing is the classification of objects in a scene. Simple examples are the classification of each character on the page of a book or the classification of a chromosome in a cell. Once the object is isolated in the scene, the goal is to describe or classify the object. If the main information for description or classification can be found in the boundary of the object it is natural to retain only the boundary for futher analysis of the object. Such situations arise, for example, in the classification of silhouettes of airplanes, photographed from the ground [8], classification of silhouettes of satellites, and in character recognition. We will restrict ourselves in this paper to the case where an object can be described by its boundary. This boundary need not be connected as for example in the character "*o*" where the boundary consists of two concentric circles. Since each boundary can be decomposed into a set of closed curves (a character *B* has for example three closed curves), we will concentrate upon the description of closed curves.

There are many techniques available to describe closed curves, but there is theoretical and experimental evidence that Fourier Descriptors are a useful set of features [1]–[3]. Other techniques are the chain encoding used by Freeman [9] and the polygonal approximation proposed by Pavlidis [12]. They encode the boundary as a sequence of curved or linear segments. The medial axis transform (MAT) uses the skeleton to describe a closed curve [4]–[6]. To describe chromosomes, Widrow [16] uses the length, width, angle, and curvature of each of the arms of a chromosome. This last technique shows that the more specific the problem is, the more efficient one can choose the descriptors of the boundary.

How do all those techniques compare? Before one can answer this one has to define the problem of boundary description and classification clearly. In its most unrestricted form, it amounts to identifying objects whose boundary might be only partially known or overlapping with other boundaries, or both. This amounts to a problem of nonideal input. Furthermore, the classes of objects might have different properties. In some cases, one needs many different prototypes in order to describe the class adequately [12]. For example, if the further processing of the boundaries does not compensate for different rotations, one might have to store several "Prototypes" corresponding to different rotations. One should realize that the problem in its most general form can, so far, only be solved by humans (and animals) and that they use *many* and *very sophisticated* analysis mechanisms using sometimes every bit of contextual knowledge to perform this task. The conclusion is that one can compare those different techniques only in the context of well-defined (simple) problems.

In this paper we will deal only with nonoverlapping boundaries that are completely known (ideal input). We will assume, furthermore, that the classes can be described by a few fixed prototypes that may be rotated, translated, scaled, and that can possess some (random) noise on the boundaries. Such a situation arises, for example, in the recognition of machine parts in automated assembly [13], [14]. The other techniques are more useful for other types of problems as will be discussed at the end of the paper.

First, we will summarize the two techniques using Fourier Descriptors [1], [2] and point out their advantages and disadvantages. Next, we give properties of the Fourier Descriptors and examine an optimal matching procedure between two closed curves. Finally, we will apply the results to a character-recognition experiment using Munson's character data set. Also, an experiment on machine parts recognition is reported. Many results given in this paper can be proved easily. For detailed information see [11].

II. Review of Fourier Descriptors

The Fourier descriptors given in [1] are defined as follows: We assume γ is a clockwise-oriented simple closed curve with parametric representation $(x(l), y(l)) = Z(l)$

Manuscript received December 8, 1975; revised July 12, 1976. This work was supported by the AFOSR under Grant 74-2661. An earlier version of this paper was presented at the Second International Joint Conference on Pattern Recognition, Copenhagen, Denmark, August 1974.

E. Persoon was with the School of Electrical Engineering, Purdue University, West Lafayette, IN 47907. He is now with Philips Research Laboratory, Eindhoven, The Netherlands.

K. S. Fu is with the School of Electrical Engineering, Purdue University, West Lafayette, IN 47907.

Reprinted from IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, vol. SMC-7, no. 3, March 1977.

where l is arc length and $0 \leq l < L$. Denote the angular direction of γ at point l by the function $\theta(l)$. Define then the cumulative angular function $\phi(l)$ as the net amount of angular bend between starting point $l = 0$ and point l. So $\phi(l) = \theta(l) - \theta(0)$ except for possible multiples of 2π and $\phi(L) = -2\pi$. Note that if γ winds in a spiral that $|\phi(l)|$ can achieve values larger than 2π. Finally, we define $\phi^*(t)$ as

$$\phi^*(t) = \phi\left(\frac{Lt}{2\pi}\right) + t \tag{1}$$

where t ranges from 0 to 2π. Note that $\phi^*(t)$ is invariant under translations, rotations, and changes of the perimeter L (scale). Note also that $\phi^*(t) \equiv 0$ for a circle. $\phi^*(t)$ is a periodic function of t if we keep scanning the boundary in a clockwise sense. Expanding $\phi^*(t)$ in its Fourier series gives

$$\phi^*(t) = \mu_0 + \sum_{k=1}^{\infty} A_k \cos(kt - \alpha_k). \tag{2}$$

The set $\{A_k, \alpha_k; k = 1, \cdots, \infty\}$ are the Fourier descriptors (FD) for curve γ.

Let us list some advantages and disadvantages of those FD's. Among the disadvantages are that no redundant information is present in the set $\{A_k, \alpha_k\}$ as will be the case for the FD's defined in [2]. Therefore, every sequence $\{A_k, \alpha_k, k = 1, 2, \cdots\}$ describes one curve and each curve has only one sequence $\{A_k, \alpha_k\}$. Among the disadvantages we have the following.

Some sequences $\{A_k, \alpha_k\}$ describe not-closed curves.

$\phi^*(t)$ for polygonal curves contains discontinuities, and therefore, the A_k will decrease rather slowly as k increases.

Reconstruction of γ requires numerical integration.

Note that we can define an equivalent set of FD's by the equation

$$\phi^*(t) = \sum_{k=-\infty}^{\infty} c_k e^{jkt} \tag{3}$$

where

$$c_k = \frac{1}{2\pi}\int_0^{2\pi} \phi^*(t) e^{-jkt}\, dt.$$

If $\{c_k\}$, $\{c_k'\}$ describe closed curves, then the coefficients $\{\beta c_k + (1-\beta)c_k'\}$, being a linear interpolation between $\{c_k\}$ and $\{c_k'\}$, do not necessarily describe closed curves.

The FD's given in [2] are defined differently. We have again that γ is a clockwise-oriented simple closed curve with representation $(x(l), y(l)) = Z(l)$, where l is the arc length along γ. A point moving along the boundary generates the complex function $u(l) = x(l) + jy(l)$ which is periodic with period L. The FD's become now

$$a_n = \frac{1}{L}\int_0^L u(l) e^{-j(2\pi/L)nl}\, dl \tag{4}$$

and

$$u(l) = \sum_{-\infty}^{\infty} a_n e^{jn(2\pi/L)l}. \tag{5}$$

We will list some advantages and disadvantages of those FD's. Among the advantages are:

All sequences $\{a_n\}$ for which the series (5) converges describe closed curves.

$u(l)$ contains no discontinuities so we can expect that $|a_n|$ decreases rather fast as $n \to \infty$.

Reconstruction of γ can be done easily.

Among the disadvantages we have the following.

$u(t)$ is a complex function, so $a_n^* \neq a_{-n}$ as in the case for real functions.

Due to the definition of $u(l)$, we have that $|du/dl| \equiv 1$ or equivalently that

$$\frac{du}{dl}\frac{du^*}{dl} \equiv 1, \quad \text{for } 0 \leq l \leq L. \tag{6}$$

If we substitute (5) into (6) it follows that there are some restrictions on the a_n. Therefore, although every sequence $\{a_n\}$ describes a closed curve using (5), not every sequence satisfies the conditions (6) and (5).

If a partial sum of (5), let us say,

$$\sum_{n=-M}^{N} a_n e^{jn(2\pi/L)l}$$

is considered, then condition (6) is equivalent to $a_n = 0$ for all $n \neq 0$ except for one value of $n \neq 0$. This shows that only circles can be described by partial sums if one insists that (6) is satisfied.

If $\{a_n\}$, $\{a_n'\}$ are two sets of FD's that satisfy (6) then the set $\{\beta a_n + (1-\beta)a_n'\}$ either satisfies (6) for all β or it satisfies (6) for no β except $\beta = 0$, $\beta = 1$. This means that this space of FD's $\{a_n\}$ that satisfy (6) and (5) is a collection of lines. Note that the curve γ described by the set $(\beta a_n + (1-\beta)a_n'\}$ is nothing but $\beta u(sL) + (1-\beta)u' \cdot (sL')$ where $0 \leq s \leq 1$ and $u(l)$ correspond to $\{a_n\}$ and $u'(l)$ corresponds to $\{a_n'\}$.

It should be noted that, if a sequence $\{a_n\}$ does not satisfy conditions (6) and (5), this means that the parameter l in (5) has the meaning of time (instead of arc length) and the boundary $u(l)$ is described with a varying velocity $|du/dl|$. So if we allow tracing of the boundary with variable speed and interpret l as time, then condition (6) is not needed.

III. Properties of the FD's

In this section we will give some more properties of the FD's. Also, a method for optimal curve matching is given.

A. Calculating FD's for a Polygonal Curve

Zahn and Roskies [1] give a formula for computing the FD's of (2) in case the curve γ is polygonal. We will give now the formula in case the FD's of (4) are used. Using the notation of Fig. 1, where V_0 is the starting point, we have

$$a_n = \frac{1}{L\left(\frac{n2\pi}{L}\right)^2}\sum_{k=1}^{m}(b_{k-1})e^{-jn(2\pi/L)l_k} \tag{7}$$

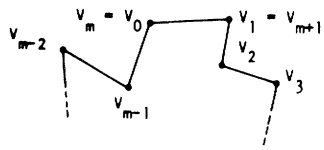

Fig. 1. A polygonal boundary.

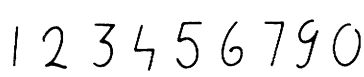

Fig. 2. Example of line patterns.

Fig. 3. Tracing of a line pattern.

where

$$l_k = \sum_{i=1}^{k} |V_i - V_{i-1}|, \quad \text{for } k > 0 \text{ and } l_0 = 0$$

and

$$b_k = \frac{V_{k+1} - V_k}{|V_{k+1} - V_k|}, \quad \text{so } |b_k| = 1.$$

B. Line Patterns

We will discuss some properties of the FD's in case a curve is not closed and nonoverlapping. Typical patterns are given in Fig. 2.

In order to be able to use the FD's, we will trace the line pattern once and then retrace it so that a closed boundary curve γ is obtained. See, for example, Fig. 3.

It can be shown that the FD's of (4) have the property

$$a_n = a_{-n} e^{-jn(2\pi/L)\alpha} \tag{8}$$

for some α, and if the path begins ($l = 0$) at one of the endpoints of the pattern, then $\alpha = 0$. Also, if (8) is satisfied for all n and some α, then any partial sum

$$u_m(l) = \sum_{-M}^{M} a_n e^{jn(2\pi/L)l}$$

describe a curve γ with the property $u(l) = u(L - l + \alpha)$ or, in other words, a curve γ that is a line pattern. Property (8) can be used, for example, to obtain skeletons of patterns as given in Fig. 4. A method of obtaining the skeleton using property (8) is given in Section IV.

In most applications the patterns will have a cetain thickness as, for example, the numerals in Fig. 4. Consider, now, the curves γ formed by the boundary of the numeral in Fig. 4. Since the starting point on γ does not carry any information for classifying those numerals, it is useful to normalize the starting point, for example, at the endpoint of the numeral (see the $\times$ in Fig. 4). This normalization is needed when we scan the field of the numeral from the left to right and top to bottom and take the

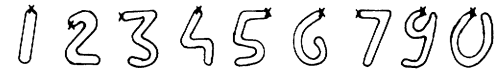

Fig. 4. Patterns with thickness.

Fig. 5. Normalized starting points.

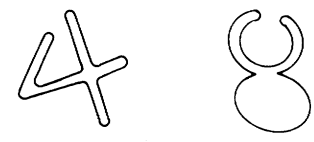

Fig. 6. Some outer boundaries.

first black point as the starting point of the boundary γ, as is shown in Fig. 5.

A convenient mathematical model of line patterns that are not closed, nonoverlapping, and have a certain thickness is (see Figs. 4 and 5)

$$u(l) = f(l) + f'(l)jc(l) \tag{9}$$

where $u(l)$ describes γ, $f(l)$ describes actually the skeleton and has the property $f(l) = f(L - l + \alpha)$, and $c(l)$ is a real positive, and periodic function giving the thickness of the pattern. For a pattern with constant thickness d, we have $c(l) = d/2$. Note that $|f'(l)| \equiv 1$. It should be noted that l is the length of the path along the skeleton not along the curve γ, and therefore the results obtained from (9) are only approximate if one insists that (6) is satisfied.

One can easily show that (for $c(l) = \text{constant} = C$)

$$a_n = b_n(1 - nw_0C)$$

$$a_{-n} = b_n e_{\alpha}^{jnw_0\alpha}(1 + nw_0C)$$

where b_n are the FD's of $f(l)$, $w_0 = 2\pi/L$, and a_n are the FD's of $u(l)$.

In order to normalize the starting point, we must have $\alpha = 0$ or

$$a_n = b_n(1 - nw_0C)$$

$$a_{-n} = b_n(1 + nw_0C). \tag{10}$$

For small values of C, the means that a_n and a_{-n} have the same phase angle. So one can normalize, for example, by finding ψ such that $a_1 e^{j}$ and $a_{-1}e^{-j\psi}$ have the same phase angle. This technique is applied in the experiment discussed at the end of this paper.

In case the curve γ cannot be modeled by (9), it becomes more difficult to normalize with respect to starting point. This is, for example, the case with the outer boundaries given in Fig. 6.

For such boundaries, one has to use the optimal curve matching technique even further in the paper.

C. Optimal Curve Matching

In many applications in pattern recognition, the position, size, and rotation of an object are not important. If we describe the object by its (outer) boundary, the starting point is irrelevant also. In nonparametric classification techniques, usually a distance is computed between the unknown sample and the nearest sample in the training set. We will use the Euclidean metric in the space of the FD's $\{a_n\}$ to compute this distance. Since position, size, rotation, and difference in the starting point are not relevant, we must scale, rotate, and adjust the starting point of each sample in the training set in order to minimize the resulting Euclidean distance. If we denote by $\{a_n\}$ the FD's of a curve α and by $\{b_n\}$ the FD's of a curve β and only M harmonics are used, then the distance $d(\alpha, \beta)$ becomes

$$d(\alpha, \beta) = \left[\sum_{n=-M}^{M} |a_n - b_n|^2 \right]^{1/2}. \tag{11}$$

Differences in position are taken out by setting $a_0 = b_0$, so we have

$$d(\alpha, \beta) = \left[\sum_{\substack{n=-M \\ n \neq 0}}^{M} |a_n - b_n|^2 \right]^{1/2}. \tag{12}$$

Suppose β is the curve in the training set so we will scale (s), rotate (ϕ), and adjust the starting point (α) such that the distance is minimized. We have then to determine s, ϕ, α such that

$$\sum_{\substack{n=-M \\ n \neq 0}}^{M} |a_n - se^{j(n\alpha + \phi)} b_n|^2 \tag{13}$$

is minimized. Note that the distance $d(\alpha, \beta)$ given in (11) also equals

$$d^2(\alpha, \beta) = \int_0^1 |u_\alpha(sL_\alpha) - u_\beta(sL_\beta)|^2 \, ds \tag{14}$$

where $u_\alpha((l)$ is the complex function decreasing α and L_α is the length of α. Therefore minimizing $d(\alpha, \beta)$ corresponds to minimizing the mean-square deviation between $u_\alpha((l)$ and $\mu_\beta(l)$.

As is shown in the Appendix, one can find the optimum values for (s, α, ϕ) by solving for the roots of a periodic function. Since this periodic function has only a finite number of harmonics, it is possible to construct a numerical technique that finds the absolute minimum for $d(\alpha, \beta)$. We will use this program in the experiment reported in Section IV.

Minimization of $d(\alpha, \beta)$ (see (14)) in the spatial domain with respect to the position, scale, and rotation is a difficult task due to the enormous number of combinations. This is especially true for boundaries that can be described sufficiently by only a few FD's, whereas a polygon or chain code approximation would require many segments to encode the boundary. However, if the curve γ is expressed in its FD's, then this problem is easily solved.

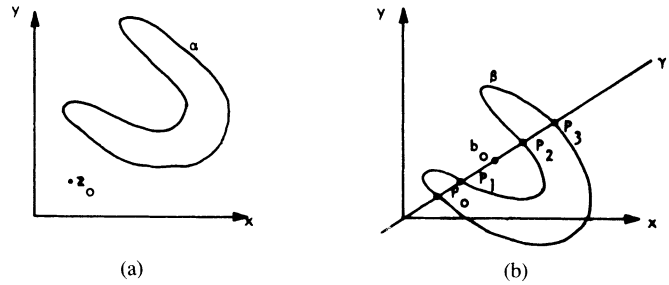

Fig. 7. A point and a boundary curve.

This is one of the nice advantages of FD's. A similar matching criterion was used by Richard and Hemami [15] and was applied to the identification of airplanes.

D. Computation of the Area of a Surface

The computation of the area of a surface given the FD's $\{a_n\}$ of its outer boundary α is a simple operation. Let us denote by $\{x(s), y(s) | s \in (0, 1)\}$ a parametric representation of the outer boundary as a function of the path length s along the boundary. S denotes the surface area. We have then

$$2S = \oint_\alpha y \, dx - \oint_\alpha x \, dy = \int_{s=0}^{1} y(s)x'(s) \, ds - \int_{s=0}^{1} x(s) \, y'(s) \, ds$$

or

$$S = -\sum_{n=-\infty}^{\infty} |a_n|^2 n\pi.$$

E. Relative Position of a Point with Respect to a Boundary

Fig. 7(a) gives a boundary curve α and a point $z_0 = x_0 + jy_0$. A useful question to ask is whether the point z_0 lies inside or outside the boundary curve α, since it provides information about the relative position of objects. If $\{a_n\}$ are the FD's of α, define then the set $\{b_n\}$ where $b_0 = a_0 - z_0$ and $b_n = a_n$ for $n \neq 0$.

The set $\{b_n\}$ describes a curve β which is a translation of α over a distance $-z_0$. The curve β is given in Fig. 7(b). After this transformation, the question can be stated equivalently as "does the origin lie inside or outside curve β." b_0 is the constant term in the series $\{b_n\}$ and denote by $cy - dx = 0$ the equation of the line γ passing through the point b_0 and the origin. We will determine the intersections of this line γ with the curve β. The intersection are given by the roots of the function $f(s)$ where

$$f(s) = c \cdot \operatorname{Im}\left[\sum b_n e^{jn2\pi s} \right].$$

Note that $b_0 = c + dj$ or $b_0^* = c - dj$.

$$f(s) = \operatorname{Re} b_0^* \operatorname{Im}\left[\sum b_n e^{jn2\pi s} \right] + \operatorname{Im} b_0^* \operatorname{Re}\left[\sum b_n e^{jn2\pi s} \right]$$

$$f(s) = \text{Im}\left[b_0^* \sum_n b_n e^{jn2\pi s}\right]$$

$$= \text{Im}\left[b_0^* b_0 + \sum_{n \neq 0} b_0^* b_n e^{jn2\pi s}\right]$$

$$f(s) = \text{Im} \sum_{n \neq 0} b_0^* b_n e^{jn2\pi s}.$$

$f(s)$ is the equation of a real function with no constant term which implies there is always at least one solution to the equation $f(s) = 0$. Assume there are m solutions $s_1, s_2, \cdots, s_m$ corresponding to points $P_1, P_2, \cdots, P_m$ on curve β. In case of Fig. 7(b), m equals 4. m is an even number unless γ is tangent to β at some point. Denote by 0 the origin. If we order the points P_i and the point 0 according to their position on the line γ, we obtain in case of Fig. 7(b), $0P_1P_2P_3P_4$. Assume m is even. The position of 0 in this rank with respect to the points P_i tells whether 0 lies inside or outside β. The rule is very simple: if there are an even number of points P_i to the left and to the right of point 0 in this ranking, 0 lies outside β; otherwise 0 lies inside β. If m is not even, the line γ might be perturbed slightly so that m becomes even. Note that $s_1, s_2, \cdots, s_m$ are the roots of a periodic function $f(s)$. They can be found using the same algorithm as mentioned in the Appendix.

IV. Experimental Results

A. Skeleton Finding Using FD's

In Section III-B, we studied the properties of FD's of nonoverlapping line patterns. We will use those properties to obtain the skeleton of such patterns. Using the mathematical model of (9) we found that

$$a_n = b_n(1 - nw_0C)$$

$$a_{-n} = b_n e^{jnw_0\alpha}(1 + nw_0C).$$

The first step of the skeleton program is to adjust the starting point such that $a_1 e^{jw_0(\alpha/2)}$ and $a_{-1}a_1 e^{-jw_0(\alpha/2)}$ have the same phase angle. So we obtain the FD's $\{a_n'\}$

$$a_n' = c_n(1 - nw_0C) = a_n e^{jnw_0(\alpha/2)}$$

$$a_{-n}' = c_n(1 - nw_0C) = a_{-n} e^{-jnw_0(\alpha/2)}$$

where $\{c_n = b_n e^{jnw_0(\alpha/2)}\}$ are the FD's of the skeleton of the line pattern.

The second step in the skeleton program is to find the coefficients c_n by the formula (15)

$$c_n = c_{-n} = \tfrac{1}{2}(a_n' + a_{-n}'). \tag{15}$$

Finally, the skeleton is obtained using (5) where a_n is replaced by c_n.

Some experimental results are given in Fig. 8. The skeletons are obtained using eight harmonics $\{a_1, a_{-1}, \cdots, a_{-8}\}$.

This method of finding skeletons is very simple but does not work, for example, on patterns as given in Fig. 6. For those patterns, another method should be developed or one can use the MAT approach as given in [4]–[6].

Fig. 8. Skeletons using eight harmonics.

B. Character Recognition

The experiments reported here are performed on the Munson data set which is a collection of Fortran symbols (numerals, characters, and special symbols) written by 49 persons. Many authors have worked with this data set before [1], [10], [12]. Each character of this set is represented by a 24 * 24 binary array. We will work only with the numerals $\{\phi, 1, \cdots, 9\}$, but the experiment can easily be performed on the entire character set.

Using a sample algorithm, the outer boundary of each numeral is obtained on a digital computer. Then, using formula (7), the FD's of each numeral are computed. Those sets of FD's will be used in all the experiments given below. The average time to find the skeleton per numeral is 12 ms and the average time to compute the FD's (15 harmonics) is 530 ms. Those times are obtained on a CDC 6500 computer.

We will give now the results obtained from three different experiments using the FD's $\{a_n\}$ as features. There are a total of 49 × 3 × 10 numerals available in the Munson set. Two training sets were used and one test set. The first training set consists of the first 160 numerals (16 samples for each type). The second training set consists of the next 310 numerals and the test set consists of the leftover 1000 numerals.

w_k \ w_i	0	1	2	3	4	5	6	7	8	9
0	39				6		1		3	
1	2	50		1	3	3	2		10	1
2	1		47						2	
3				46						
4	1				27				2	1
5						47			2	
6	1		1		2		47		2	
7	1		2	1	1			50		2
8	4				4				26	1
9	1			2	7				3	45

error rate = $\frac{76}{500}$ = 15.4%

Fig. 9. Classification result (suboptimal procedure).

w_k \ w_i	0	1	2	3	4	5	6	7	8	9
0	42				4		1		1	
1	1	50			1	2	2		10	1
2	1		47						2	
3				47						
4	1				43	1	1		3	2
5						47			2	
6			1				46		1	
7			2	1				50		2
8	4								31	1
9	1			2	2					44

w_i original class

w_k class classified to

error rate = $\frac{53}{500}$ = 10.6%

Fig. 10. Classification result (optimal procedure).

1) Suboptimal (Fixed Sample Size): In this experiment, each numeral is normalized for scaling, rotation, position, and starting point. In other words, $a_0 = 0$ and a_n is multiplied with $se^{j(\phi + n\alpha)}$ for $n \neq 0$. The parameters s, ϕ, and α are chosen such that a_1 and a_{-1} become pure imaginary numbers and such that $|a_1 + a_{-1}| = 1$. The first feature $x_1 = |a_1 - a_{-1}|$ and $x_2 = \text{Re}\ (a_2)$, $x_3 = \text{Im}\ (a_2)$, $x_4 = \text{Re}\ (a_{-2})$, $x_5 = \text{Im}\ (a_{-2})$, $x_6 = \text{Re}\ (a_3)$, and so on. Since 8 harmonics are used in this experiment, a total of $1 + 4 \times 7 = 29$ features is available.

In order to classify an unknown numeral, the distance between that numeral and each numeral in the training set is computed. The distance used here is

$$\left[\sum_{i=1}^{29} (x_1 - x_i')^2\right]^{1/2} \tag{16}$$

where $\{x_1\}$ and $\{x_i'\}$ are the features corresponding to the numerals between which the distance is computed. This distance is equivalent of $d(\alpha, \beta)$ given (12) except for the first harmonic. The unknown numeral is then given the same class label as the label of the closest numeral in the training set.

In the first phase of the experiment, the first training set (160 samples) was taken and the test set consisted of the second training set (310 samples). Based upon those results, a new training set of 160 samples was constructed using numerals of both training sets. Then the first 500 samples of the test set were processed and the results are given in Fig. 9. The total error rate is 15.4 percent. Note that only 500 samples were processed in order to save computer time. The time required to process one numeral is 150 ms (CDC 6500).

In Fig. 9 we observe that numerals "4" and "8" are the most difficult ones to classify. The reason for this is that the normalization technique that determines s, ϕ, and α is not good for the numerals "4" and "8" as is shown in Section III-B and Fig. 6.

2) Optimal Procedure (Fixed Sample Size): In this experiment we used the same modified training set as in the previous experiment and also the same 500 test samples. The only difference is that the distance of (13) is used instead of the distance of (16). The results of this experiment are given in Fig. 10. We see that numeral "4" is much better recognized. Only "8" is misclassified many times. We may not forget however that only the outer boundary is used so that "1" and "8" look very similar. The time required to process one numeral (eight harmonics) is 14 s on a CDC 6500 and 240 s on a CDC 1700. The total error rate for the experiment is 10.6 percent.

3) Suboptimal (Variable Sample Size): Since a skeleton is obtained in 12 ms and the computation of one harmonic (a_n, a_{-n}) requires $530/15 = 35$ ms we conclude that most of the computation time goes to the computation of the FD's $\{a_n\}$. In order to minimize the computation time, a sequential decision algorithm is used in this experiment.

The experiment is identical to the experiment in Subsection B-1) except for the following. Each time another coefficient a_n of the unknown numeral is computed (using (7)), one computes the distance (16) between that numeral and each numeral in the training set. One finds then the distances $\{d_k | k = 0, \cdots, 9\}$ where d_k is the smallest distance between the unknown numeral and the 16 numerals "k" in the training set. Then one computes

$$p_k = \frac{1}{16(d_k)^M}$$

where M is the number of terms used in (16). Note that p_k is actually an estimate of the conditional probability density. One computes then

$$r = 1 - \max_i \frac{p_i}{\sum p_k}.$$

This number r is an estimate of the conditional probability of error for this unknown sample. If r is less than a certain threshold e^* one stops and classifies the numeral to the class k where $p_k = \max_i p_i$. If r is larger than e^*, one computes another coefficient a_n using (7) and repeats the process. The threshold e^* can be chosen and controls the error rate.

The results of this experiment are given in Figs. 11 and 12. In this experiment, the entire test set of 1000 samples was used and three different cases were investigated. In case A, a maximum of 9 harmonics (33 features) was al-

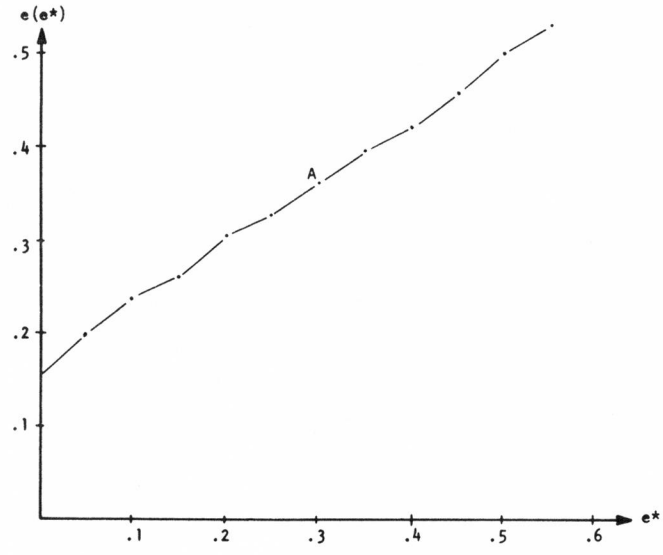

Fig. 11. Error rate as a function of threshold.

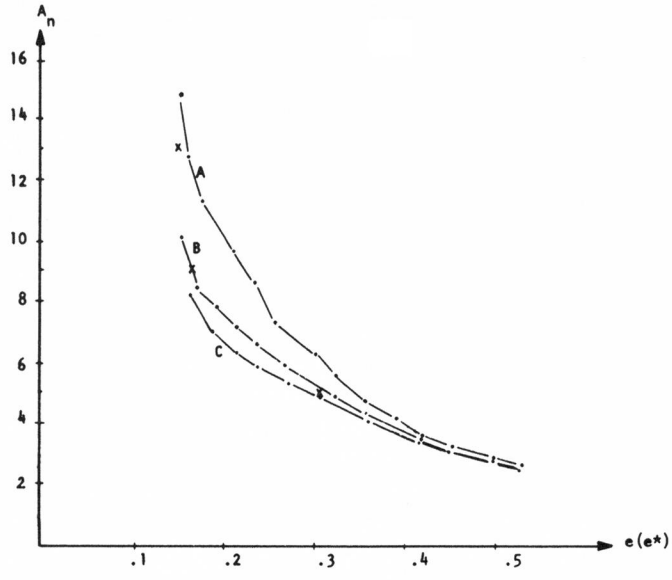

Fig. 12. Average number of features as a function of error rate.

lowed. If then the procedure asks for more features (X_{34}), the procedure stops and classifies the numeral. In cases B and C, a maximum of 4 (13 features) and 3 (9 features) harmonics, respectively, were allowed.

Fig. 11 gives the error rate $e(e^*)$ as a function of the threshold e^* for case A (the curves for cases B and C fall very close to the curve for case A and are, therefore, not drawn). It shows that the threshold e^* nicely controls the error rate $e(e^*)$.

Fig. 12 gives the average number of features A_n needed for classification as a function of the error rate $e(e^*)$ for cases A, B, and C. The three points (marked with $\times$) are obtained from fixed sample size experiments (see Subsection C-1)) using 5, 9, and 13 features, respectively. Compare now the top of curve *B* and the point corresponding to the fixed sample size procedure using 13 features. Both points have the same error rate, but the variable sample size procedure requires only 10.1 features on the average. The same holds for the top of curve *C* and the fixed sample size point for 9 features. This shows the usefulness of the variable sample size approach.

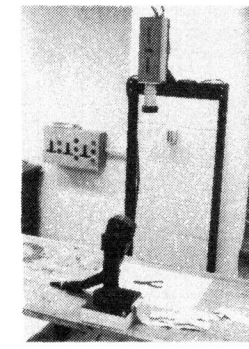

Fig. 13. Manipulator and television camera.

C. Machine Parts Recognition

Recently, much research has been done on visual feedback to aid in computer control of manipulators (mechanical arms) [13], [14]. The visual information is usually obtained from one or more television cameras or linear diode array cameras. The cameras are aimed at a working area which is a horizontal flat surface (table top or moving belt) and which is accessible by the manipulator (see Fig. 13). Both manipulator and cameras are interfaced to a minicomputer.

The major applications of using manipulators is either in assembly tasks [13], [14] or in inspection tasks [14]. In assembly tasks, the working area contains several (overlapping) parts and the manipulator is supposed to assemble them into a product. In inspection tasks, the computer examines the parts, using visual feedback, and decides which ones are defective. The manipulator then separates good and defective parts. Another application [14] is the packing or unpacking of parts into boxes.

The visual feedback contains two parts: picture aquisition and picture recognition. The output of the aquisition part is a matrix of intensity values representing the input scene. In the acquisition phase it is important to obtain a digitized image which makes the recognition task as easy as possible. Appropriate lighting, viewing angle, color filters, and special backgrounds are helpful in this aspect. In order to reduce the number of different views one can obtain from an object, the working area is a horizontal surface and the camera is mounted vertically high enough above the working area. Due to gravity, the object usually has only a small number of distinct stable positions on such a surface and, therefore, for a given stable position, the camera will see always the same scene except for translation, rotation, and scaling (in case a zoom is available). In most instances, the silhouette of an object is adequate for recognition. Using dark objects on a white background or vice versa, backlighting, fluorescent surfaces, and so on, and also using a good quality television camera, it is possible to obtain the silhouettes of parts by

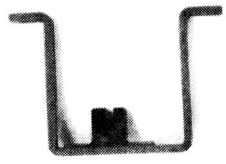

Fig. 14. U-shaped part.

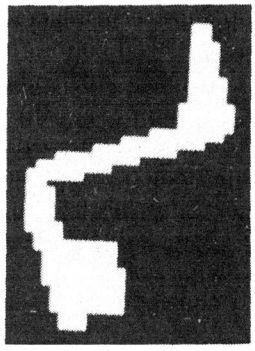

Fig. 15. Silhouette of S-shaped part.

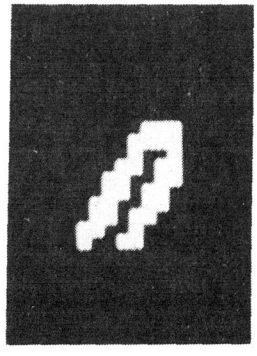

Fig. 16. Silhouette of U-shaped part.

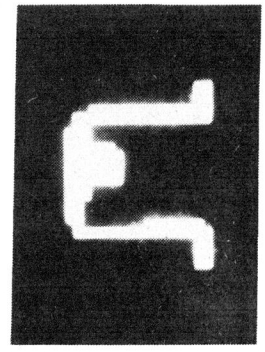

Fig. 17. Silhouette of U-shaped part.

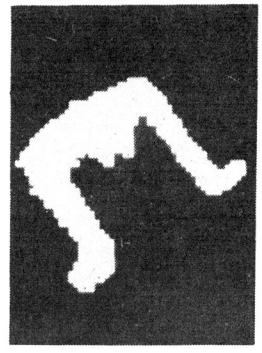

Fig. 18. Silhouette of U-shaped part.

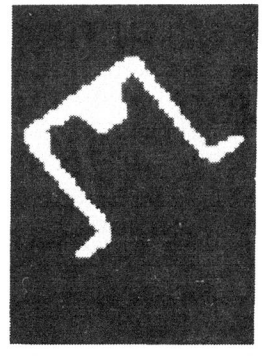

Fig. 19. Silhouette of U-shaped part.

simple thresholding (see Figs. 14–20). The output of the aquisition part is then a matrix of binary values.

The picture recognition part is supposed to recognize the object using the information contained in this binary matrix. Several factors can make the recognition difficult. Depending on the value of the threshold, the silhouette can have a different thickness (see Figs. 18 and 19 which are obtained from the object shown in Fig. 14). Another problem is the finite resolution of the aquisition system. Especially for small objects, this causes significant quantization noise at the boundary (see Fig. 16). Those and other factors make recognition a nontrivial task.

In the experiment reported here we use a commercial camera mounted vertically above a table (Fig. 13). The camera is connected to a PDP 11/45 computer using an interface that digitizes one vertical line of the input scene in 1/60 s. A vertical line is digitized into 250 pixels with

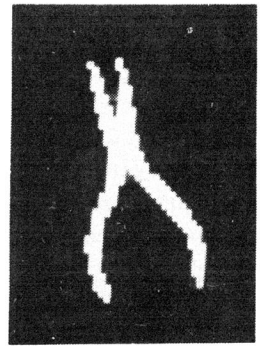

Fig. 20. Silhouette of pliers.

7 bits of accuracy (128 intensity levels). In the experiment we use only 60 of the 250 pixels and scan 60 vertical lines giving a 60 by 60 matrix of pixels. The gray level histo-

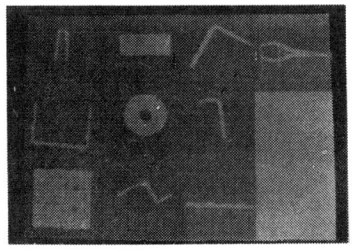

Fig. 21. Training set.

gram of this matrix is computed and used to determine a threshold. This threshold allows us to convert the matrix into a binary matrix representing the silhouette of the part. The input data are now in the same format as the Munson data set. The recognition part is the same as that used in Section B-2).

The recognition scheme, implemented for ten classes as shown in Fig. 21, requires on the average 11 s per unknown object of which 1 s is needed to scan the image with the television camera. For parts recognition, one also needs an algorithm that gives the rotation and scale of the object since those need to be known in order to group the object with a manipulator. Rotation and scale can be found using (A1) and (A2) in the Appendix. The training set consists of the silhouettes of 10 different parts (Fig. 21). Although some of the parts have more than one stable position, only one position for each object was recorded in the training set. The performance of the algorithm was very good. One could characterize the performance by an error rate on a test set but this would be meaningless without publishing the test set also. Therefore, we will give some examples which give an idea of the variety allowed in the input patterns. The algorithm classifies the object in Fig. 15 to class 6, that in Fig. 16 to class 1, those in Figs. 17–19 to class 2, and the object in Fig. 20 to class 10. The object shown in Fig. 15 has a protrusion at the end not present in the shape of class 6. The pliers in Fig. 20 are opened while they are closed in the shape of class 10. This shows that minor variations of the input patterns are allowed. Rotation, translation, and scaling of the object do not influence the performance of the algorithm.

The advantage of this approach to "parts recognition" is that it is general and does not require human interaction or special software when other parts have to be recognized. Also the optimal matching technique is simple and fast in execution.[1] The approach taken by Ambler *et al.* [13] can work on boundaries of silhouettes in which some segments are missing. Their algorithms are, however, much more complex and need more training time. The approach taken at SRI [14] requires interaction of the operator since features are extracted from the silhouettes and appropriate features have to be selected for each new set of parts. In our experiment, only the outer boundary of the part is taken into account. The algorithm can be improved, however, by also considering possible holes that can appear in the part [13]. Overlapping parts can be detected as follows: either the boundary obtained from overlapping parts does not match closely enough with one of the training patterns ($d(\alpha, \beta)$ in (13) is too large) or, in case it matches closely, the area of the silhouette is not as expected from a single object. In such a case, the arm will try to separate the parts. This approach is taken by Ambler *et al.* [13].

[1]Using (13) with $M = 15$.

V. Discussion

In this paper, the FD's were computed using (4) or (7). In (4) one can also interpret the parameter l as time and scan the boundary with uniform velocity. It is possible then to study the effect of moving along the boundary with nonuniform velocity. It is shown in [18] that, when appropriate velocity patterns are chosen, it requires fewer harmonics to store the boundary curve accurately.

From the experimental results in Section IV, it is evident that FD's are very useful for the problems defined in the Introduction mainly because they allow an easy recognition for rotated and scaled patterns having some noise on their boundaries. Other techniques, as polygonal approximation, are not so good for this type of problem since compensation for rotation is not easy in such a description. Polygonal approximation and chain encoding should be used in other types of problems where the patterns are best described in terms of specific features present in the boundary (e.g., number of concave arcs in the boundary). Pavlidis presents such a list that is useful in character recognition [12]. The main problem with such techniques [12], [13], [15], [16] is that, due to the state of the art, there are no standard techniques that allow the user to find the useful features or to select them. It is typical that in applying such a technique, the user has to be quite familiar with the problems. This is not necessarily the case when using the FD's.

Appendix

Optimal Curve Matching

Consider two curves α and β with FD's $\{a_n\}$ and $\{b_n\}$, respectively. Define the distance $d(\alpha, \beta)$ between α and β as in (20). One wants to scale (s), rotate (ϕ), and adjust the starting point (α) of curve β such that $d(\alpha, \beta)$ is minimized. We have then to determine s, ϕ, α such that

$$\sum_{\substack{n=-M \\ n\neq 0}}^{M} |a_n - se^{j(n\alpha+\phi)}b_n|^2$$

is minimized. This expression equals

$$\sum [a_n - se^{j(n\alpha+\phi)}b_n]\,[a_n^* - se^{-j(n\alpha+\phi)}b_n^*]$$

$$= \sum a_n a_n^* + s^2 \sum b_n b_n^* - 2s \sum \text{Re}\,[a_n^* b_n e^{j(n\alpha+\phi)}].$$

Set $a_n^* b_n = \rho_n e^{j\psi n}$. Then we have, equivalently,

$$\sum a_n a_n^* + s^2 \sum b_n b_n^* - 2s \sum \rho_n \cos(\psi_n + n\alpha + \phi).$$

In order to minimize this expression with respect to s, α, ϕ we will compute the partial derivatives.

$$\frac{\partial}{\partial s} = 2s \sum b_n b_n^* - 2 \sum \rho_n \cos(\psi_n + n\alpha + \phi)$$

$$\frac{\partial}{\partial \phi} = 2s \sum \rho_n \sin(\psi_n + n\alpha + \phi)$$

$$\frac{\partial}{\partial \alpha} = 2s \sum \rho_n n \sin(\psi_n + n\alpha + \phi).$$

By setting those derivatives equal to zero we obtain

$$s = \frac{\sum \rho_n \cos(\psi_n + n\alpha + \phi)}{\sum b_n b_n^*} \tag{A1}$$

$$\tan \phi = -\frac{\sum \rho_n \sin(\psi_n + n\alpha)}{\sum \rho_n \cos(\psi_n + n\alpha)} \tag{A2}$$

$$\tan \phi = -\frac{\sum \rho_n n \sin(\psi_n + n\alpha)}{\sum \rho_n n \cos(\psi_n + n\alpha)}. \tag{A3}$$

Combining (A2) and (A3) we obtain an equation in α

$$f(\alpha) = \sum \rho_n \sin(\psi_n + n\alpha) \sum n\rho_n \cos(\psi_n + n\alpha) - \sum \rho_n \cos(\psi_n + n\alpha) \sum n\rho_n \sin(\psi_n + n\alpha).$$

The optimum value for α is then a zero of $f(\alpha)$ and the corresponding values for ϕ and s are obtained from (A2) and (A1).

Since $f(\alpha)$ contains only a finite number (M) of harmonics, it is possible to find all the zeros of $f(\alpha)$ using numerical techniques. A subroutine was written that finds *all* the roots up to machine accuracy.

References

[1] C. T. Zahn and R. Z. Roskies, "Fourier descriptors for plane closed curves," *IEEE Trans. Computers*, vol. C-21, pp. 269-281, 1972.

[2] G. H. Granlund, "Fourier preprocessing for hand print character recognition," *IEEE Trans. Computers*, vol. C-21, pp. 195-201, 1972.

[3] E. L. Brill, "Character recognition via Fourier descriptors," presented at WESCON, Session 25, Qualitative Pattern Recognition Through Image Shaping, Los Angeles, Calif., Aug. 1968.

[4] H. Blum, "A transformation for extracting new descriptors of shape," in *Models for the Perception of Speech and Visual Form*, W. Wathen-Dum, Ed. Cambridge, Mass.: MIT Press, 1967.

[5] O. Philbrick, "Shape description with the medial axis transformation," in *Pictorial Pattern Recognition*, G. C. Cheng, Ed. Washington, D.C.: Thompson, 1968, pp. 395-407.

[6] J. C. Moth-Smith, "Medial axis transformations," in *Picture Processing and Psychopictorics*, Lipkin and Rosenfeld, Eds. New York: Academic Press, 1970, pp. 267-283.

[7] G. S. Sidhu and R. T. Boute, "Property encoding: Application in binary picture encoding and boundary following," *IEEE Trans. Computers*, vol. C-21, pp. 1206-1216, Nov. 1972.

[8] R. B. NcGhee, "Automatic recognition of complete three-dimensional objects from optical images," presented at the Second U.S.-Japan Seminar on Learning Control and Intelligent Control, Gainesville, Fla., Oct. 22-26, 1973.

[9] H. Freeman, "Boundary encoding and processing," in *Picture Processing and Psychopictorics*, Lipkin and Rosenfeld, Eds. New York: Academic Press, 1970, pp. 241-306.

[10] A. B. S. Hussain, G. T. Toussant, and R. W. Donaldson, "Results obtained using a simple character recognition procedure on Munson's handprinted data," *IEEE Trans. Computers*, vol. C-21, pp. 269-281, 1972.

[11] E. Persoon and K. S. Fu, "Sequential decision procedures with specified error probabilities and their applications," Rept. TR-EE74-30, Purdue University, 1974. Also published in "Interdisciplinary Systems Research" Series, Birkhauser Verlag, Elisabethenstrasse 19, CH-4010 Basel, Switzerland.

[12] T. Pavlidis and F. Ali, "Computer recognition of handwritten numerals by polygonal approximations," *IEEE Trans. Systems, Man, Cybern.*, vol. SMC-6, pp. 610-614, Nov. 1975.

[13] A. P. Ambler et al., "A versatile system for computer controlled assembly," *Artificial Intelligence*, vol. 6, pp. 129-156, 1975.

[14] C. Rosen *et al.*, "Exploratory research in advanced automation," Stanford Res. Inst., Aug. 1974.

[15] C. W. Richard and H. Hermami, "Identification of three-dimensional objects using Fourier descriptors of the boundary curve," *IEEE Trans. Systems, Man, Cybern.*, vol. SMC-4, pp. 371-378, July 1974.

[16] B. Widrow, "The rubber-mask technique," *Pattern Recogn.*, vol. 5, pp. 175-197, 1973.

[17] E. Persoon, "Object recognition using Fourier descriptors," presented at Milwaukee Symposium on Automatic Control, Milwaukee, Wisc., Apr. 22-24, 1976.

[18] E. Persoon and K. S. Fu, "Shape discrimination using Fourier descriptors," presented at Second International Joint Conference on Pattern Recognition, Copenhagen, Denmark, Aug. 1974.

[19] J. H. Munson, "Experiments in the recognition of handprinted text: Part I—Character recognition," in *1968 Fall Joint Comput. Conf. AFIPS Conf. Proc.*, vol. 33. Washington, D. C.: Thompson, 1968, pp. 1125-1138.