A METHOD FOR INARIANT PATTERN RECOGNITION BY USING THE SCALE-VECTOR REPRESENTATION OF PLANAR CURVES *

J.A. García, J. Fdez-Valdivia and R. Molina
Departamento de Ciencias de la Computación e I.A.
E.T.S. de Ingeniería Informática.
Universidad de Granada. 18071 Granada. Spain.

Keywords and phrases: Curve segmentation, clustering, single scale, Gaussian smoothing, zeros of curvature, dominant points, polygonal approximation, turning function.

Corresponding author: J.A. García
Departamento de Ciencias de la Computación e I.A.
E.T.S. de Ingeniería Informática.
Universidad de Granada. 18071 Granada. Spain.
e-mail : jags@robinson.ugr.es
fax : +34.958.243317

Complete address for all authors, and mailing address for galley proofs:
Departamento de Ciencias de la Computación e I.A.
E.T.S. de Ingeniería Informática.
Universidad de Granada. 18071 Granada. Spain.

Number of manuscript pages: 28
Number of figures: 8
Number of tables: 2

*This work has been supported by the Spanish Dirección General de Ciencia y Tecnología (DGCYT)
under Grant PM-0093-C02-02
Abstract
This article presents a new process for shift, rotation, and scale invariant pattern recognition using shapes. We represent the shape by using its two-dimensional contour, and we describe this planar curve not at all different scales, but each one of parts in the curve isolating a different structure at a single scale. A scale-vector representation of contours usually avoids missing fine features and overlooking coarse features. A polygonal approximation of these planar curves can be made by joining the successive dominant points detected on the contour represented at its scale vector. Dominant points of digitized curves are points with high curvature value. We present a scale-vector based dominant point detection algorithm which needs no input parameter and remains reliable even when features of multiple size are present on the digital contours. Model-based recognition is achieved by comparing the polygonal approximation to the contour extracted from shape A, which is stored as a model for some particular object, with the polygonal approximation to the contour extracted from shape B, which is found to exist in an image. To compare polygons we use the $L_2$ distance between the turning functions of the two polygons. This method to compare polygons is invariant under translation, rotation and change of scale, taking time $O(mn \log mn)$ to compare an $m$ vertex polygon against an $n$ vertex polygon.

1 Introduction

As a result of the recent advances in image acquisition hardware and increased computational power of computers, at present object classifying by pattern recognition techniques becomes possible. The recognition of objects is done by using shape, texture, colour, gray scale, and so on. In this work only shape was analysed. More precisely, the process of comparing and recognizing shapes is accomplished by analysing the information of the shapes’ boundaries. The problem of shape recognition has received considerable attention in the fields of computer vision, pattern analysis and object classification (Pavlidis[17]). Shape recognition is an easy task for humans, but it is still quite difficult for computers. Because an object can be rotated or translated, the problem of shape recognition becomes more and more complicated.

There are many approaches for solving pattern recognition problems, categorized into two main types: syntactic and statistical (Fu[5], Mantas[13]). In the first, a shape is de-
composed into some basic primitives, which are represented by a sentence in a language specified by a grammar, and then a LR parser is used for recognizing shapes. Because the parsing table and grammar are complex as well as computationally expensive, syntactic methods have limited applicability. In the statistical approaches, attribute evaluation and selection using a performance metric related to the classifier’s probability of error is necessary to identify effective feature sets. A statistical method of applying polar signatures to represent and match shapes is proposed in Lie[10]. However, larger storage capacity is always needed for polar signatures.

The purpose of this paper is to investigate how to overcome the disadvantages mentioned above. A simple method for shape recognition is proposed based upon polygonal approximations. Such representations not only substantially reduce the volume of data to be processed but also achieve significant unification of object shapes. However, what is important and new about our approach is that different parts of the boundary curve are represented at different scales as appropriate to the particular situation on that part of the boundary—i.e. the degree of local smoothing varies with the local shape characteristics. We assume therefore a very realistic assumption, that is the planar curve has several non-overlapping shape characteristics, and each one of them can be isolated in a different part of the curve; and so to describe the curve, it is necessary that each one of the curve parts isolating a different characteristic be described at its degree of smoothing (the scale of the shape feature being isolated in the curved part). This assumption is verified by many curves: cartographic boundaries (Garcia[7]), biological contours (Fdez – Valdivia[4], Pérez de la Blanca[18]), frontiers extracted on astronomical images (García[6]), etc.

To represent the planar curve this paper describes a novel solution, the scale-vector representation, in which it is guaranteed that each structure will be represented at its single scale, avoiding to represent the different shape characteristics at all the discrete scales between a fixed range, as multi-scale methods perform (Lowe[11], Bengtsson[3], Mokhtarian[14], Rattarangsi[19], Rosin[23]). By using a single scale at each position on the boundary, savings in computation are achieved relative to an uniform multi-scale representation of the entire boundary. Also, our representation of contours usually avoids missing fine features and overlooking coarse ones.
A polygonal approximation of these planar curves can be made by joining the successive dominant points detected on the contour represented at its scale vector. Dominant points of digitized curves are points with high curvature. Such points, as observed by Attneave\cite{2}, contain important information about the shape of an object. We present a procedure for detecting dominant points which needs no input parameter and remains reliable even when features of multiple size are present on the digital contours.

Model-based recognition is achieved by comparing the polygonal approximation to the contour extracted from shape \( A \), which is stored as a model for some particular object, with the polygonal approximation to the contour extracted from shape \( B \), which is found to exist in an image. If \( A \) and \( B \) are close to being the same shape, then our vision system reports a match and returns a measure of how good that match is. To compare polygons we use the \( L_2 \) distance between the turning functions (\( Arkin^{[1]} \)) of the two polygons. This method to compare polygons is invariant under translation, rotation and change of scale.

In Section 2, we give a representation for 2-D contours at their scale vectors \( (\sigma^1, \cdots, \sigma^L) \), being the planar curve partitioned into \( L \) parts \( C_1, \cdots, C_L \) each one of them isolating a different structure, and with \( \sigma^i \) being the natural scale to describe the level of structure in \( C_i \). In Section 3, we present a procedure for detecting dominant points which needs no input parameter and does not involve determination of a region of support, and we obtain a polygonal approximation of the planar curve by joining the successive dominant points detected to the contour represented at its scale-vector. In Section 4, we describe how the two-dimensional contours encoded by their polygonal approximations may be directly compared with similarly encoded patterns stored in memory, allowing them to be recognized. Experimental results are presented in Section 5. Section 6 presents the conclusions of this paper.

2 Representing planar curves by using a scale vector

We assume that many curves have structures at a variety of different scales and therefore there can be no single level of smoothing. As, in general, the scales required are not known, one solution to the representation problem is to represent the curve at multiple scales, so that each structure can be represented at its appropriate scale. Most multi-scale methods
represent the features at all the discrete scales between a fixed range. This guarantees that each any structure appearing on the contour will be represented in the right scale. However, there will be little qualitative change between most curves at adjacent scales, and so much of the data is redundant. Such representations are cumbersome as the most interesting information is not made explicit, and it is inefficient to have to process all the curves for the subsequent stages of matching and recognition.

To solve the representation problem, we describe the contour \( C \) not at all different scales \( \sigma \), but each part \( C_i \) of \( C \) showing a single structure at its scale \( \sigma^i \). Therefore, we represent planar curve at its scale vector \( (\sigma^1, \cdots, \sigma^L) \), assuming that the curve is partitioned in \( L \) parts \( C_1, \cdots, C_L \), each one of them isolating a single structure.

### 2.1 Curve segmentation

To divide the contour \( C = \{p_i = (x_i, y_i) \mid i = 1, \cdots, n\} \) into its parts (each one generated by a different process), we segment the 2-D curve using a clustering procedure. The objective of clustering is to divide the curve into parts in a way that will reveal any underlying structure. The clustering algorithm used in this paper requires neither the number of clusters in the data nor the minimum level of homogeneity for the entities within a cluster.

We cluster the 2-D curve \( C \) by using different statistical measures to identify features on the graph of the curvature values \( K = \{k_i \mid i = 1, \cdots, n\} \) computed from the digital curve \( C = \{(x_i, y_i) \mid i = 1, \cdots, n\} \), and \( k_i \) being the curvature value of \((x_i, y_i)\). The continuous expression to calculate curvature values on each point of a parametric curve \( \{(x(t), y(t) \mid t \in R)\} \) is

\[
k = \frac{\ddot{x} \ddot{y} - \dot{x} \dot{y}^2}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \tag{1}
\]

where \( (\cdot)' \), \( (\cdot)'' \) denote first and second derivatives respectively. To apply this expression to our discrete contour \( C \) we must control the spatial quantization noise. To reduce the noise, we have estimated the curvature value at each point by using a Gaussian filter, with \( \sigma = 1 \), to smooth the quantization noise (Mokhtarian[14]).

The statistical measures used to identify features on the curve are based on the absolute value of differences in curvatures \( k_i \). More precisely, the measures are based on the 2-D histogram representing the number of occurrences in the curve of the absolute value of
differences in curvatures over a distance $\delta$. The histogram is calculated at each sub-interval $[(i - 1)s, is)$ (with $s = \frac{2\pi}{360}$ and $i = 1, \cdots, 360$).

We assigned $\text{dif}(i, \delta)$ to the absolute value of difference in curvature on two points of the curve at a distance $\delta$ ($\delta$ integer)

$$\text{dif}(i, \delta) = |k_i - k_{i+\delta}|$$

and $H(j, \delta)$ to the probability that the absolute value of one difference in a particular sub-interval $[j-1)s, js)$, with $j \in \{1, \cdots, 360\}$, will occur according to the value of displacement $\delta$. The measures we considered from $H$ were:

$$\eta_{\delta} = \sum_{i=1}^{360} \frac{(2\pi)^2}{360} \times H(i, \delta)$$

$$\beta_{\delta} = \sum_{i=1}^{360} \frac{(2\pi)^2}{360} \times H(i, \delta)$$

$$E_{\delta} = - \sum_{i=1}^{360} H(i, \delta) \times \log H(i, \delta)$$

$$A_{\delta} = \sum_{i=1}^{360} H(i, \delta)^2$$

$\eta_{\delta}$ corresponding to an average of the absolute value of the differences; $\beta_{\delta}, A_{\delta}$ can be considered measures of strength of the absolute value of differences, and $E_{\delta}$ of uniformity.

### 2.2 Clustering

In the clustering process, the given curve is initially divided into a number of nonoverlapping segments $C_i$, where the segment size is selected to produce homogeneous segments containing enough information about their underlying structure (García[81]).

On an experimental level, we observe that the partition process is robust for reasonable elections of the value of displacement $\delta$; and the values of 2 and 4 for $\delta$ were therefore chosen empirically.

For each segment $C_i$ in the partition, a vector $\Phi^i = (\phi_1^i, \phi_2^i, \cdots, \phi_n^i)$ is calculated, where

$$(\phi_1^i, \phi_2^i, \cdots, \phi_n^i) = (\eta_2, \beta_2, E_2, A_2, \eta_4, \beta_4, E_4, A_4).$$
A distance measure for these vectors is defined as:

\[ dist(C_i, C_j) = \max_i \left\{ \left| \phi_i^1 - \phi_i^j \right| \right\} \] (6)

A simple clustering is done based on this distance measure. First the minimum distance between any two parts is found as

\[ dist(C_{i\ast}, C_{j\ast}) = \min_{i,j} dist(C_i, C_j) \] (7)

Then the nearest pair of parts \( C_{i\ast}, C_{j\ast} \) at the level are to be fused to one at the next hierarchical level.

2.3 The number of curve parts

The number of parts, \( N^\ast \), is unknown, and we need to obtain a suitable value of \( N^\ast \), with the help of an optimizing criterion for merging parts. Our approach to solve this problem, is to consider the between-parts difference plotted against the number of parts at different levels of the hierarchical clustering, assuming that the largest growth in this plot indicates the number of parts in the given curve.

Let \( d(t) \) be the dissimilarity measure at level \( t \) of the hierarchical clustering, defined as

\[ d(t) = dist(C_{i\ast}, C_{j\ast}) \]

with \( C_{i\ast}, C_{j\ast} \) being the nearest pair of parts at the level \( t \) that are to be fused to one at the next level \( t - 1 \).

The dissimilarity measure, in terms of the minimum of between-parts distances, is used to give an estimate of the number of clusters \( N^\ast \) as

\[ N^\ast = \arg \max_t \left\{ \left| d(t) - d(t + 1) \right| \right\} \] (8)

The value \( N^\ast \) corresponds to the level at which the largest growth of the dissimilarity measure occurs, which we assume suggests the presence of natural groupings. This rule to determine the number of parts is objective in the sense that it assumes no threshold, and as a consequence leads to an objective selection of \( N^\ast \). The rule considered here also makes the problem of the partition of the given curve into an unknown number of parts well-defined.
Once a partition to $C$ is obtained, each cluster of the partition determines one curve part $C_i$ of the vector $(C_1, \ldots, C_L)$, dividing the contour $C$ at parts revealing different underlying structures.

2.4 A scale-vector representing the planar curve

We have partitioned the curve $C$ into a number of curve parts $C_1, \ldots, C_L$, where each one of them shows a single feature. We now provide a curve representation using a scale-vector $(\sigma^1, \ldots, \sigma^L)$, with $\sigma^i$ being the single scale to $C_i$.

We will use Gaussian smoothing to obtain a representation of the curve part $C_i$ at a specific scale. To smooth the curve part, it is convolved with a one-dimensional Gaussian Kernel $G_\sigma(t)$ of standard deviation $\sigma$, using the technique described in Mokhtarian\cite{14}.

The curved part $C_i$ is smoothed over a range of octave separated scales, with $\sigma$ starting at 1 and increasing by a factor of $\sqrt{\sigma}$. The maximum amount of smoothing required to locate the coarsest structure will depend on the size and shape of the curve. The shorter and smoother the curve part $C_i$ is, the less smoothing is required.

We will choose a normalized measure of the zeros of curvature of $C_i$ at a scale $\sigma$

$$Z_\sigma(C_i) = \sigma \sum_{i \{x_i, y_i\} \in C_i} z(k_i),$$

(9)

with

$$z(k_i) = \begin{cases} 1 & \text{if } k_i = 0 \\ 0 & \text{otherwise} \end{cases},$$

as a significance measure describing fundamental features of the curve’s shape (Rosin\cite{22}).

To find out a single scale for $C_i$, we choose the scale minimizing this measure. Minima of this significance measure over different scales correspond to structure in the curve having been isolated at its natural scale. A discussion about this measure can be found in Rosin\cite{22}.

Let $\sigma^i$ be the single scale to $C_i$, which is defined as the value of $\sigma$ verifying

$$Z_{\sigma^i}(C_i) = \min_\sigma Z_\sigma(C_i).$$

(10)

Then we describe each curve part $C_i$ of $C$ at its single scale $\sigma^i$, with $i = 1, \ldots, L$, and we represent the planar curve $C$ at the scale vector $(\sigma^1, \ldots, \sigma^L)$. 

8
3 Detection of dominant points and polygonal approximation of digitized contours

We now present an algorithm for detecting dominant points of digitized curves. A polygonal approximation of curves is made by joining the successive dominant points.

Single-scale dominant point detection algorithms are unreliable because objects, in general, cannot categorically be assumed to have only features of a single size. Dominant point detection must therefore observe these points at the most appropriate scales.

We solve the problem of dominant point detection on digital boundaries, representing each contour at its scale vector. The idea is based on the argument that curvature measurements of an contour at its scale vector contains important information about physically significant features.

In this section, we present a scale-vector based dominant point detection algorithm that uses a scale-vector representation of the boundary. Let $C = \{(x_i, y_i) \mid i = 1, \ldots, n\}$ be the sequence of $n$ integer-coordinate points describing a non-self-intersecting curve in a rectangular grid. To find the set of points characterizing $C$’s perceptual shape we use the graph of its curvature values calculated at its scale-vector.

Let $K(\sigma^1, \ldots, \sigma^L) = \{k_i(\sigma^j) \mid i = 1, \ldots, n\}$ be the curvature values computed from the digital curve $C$, where $k_i(\sigma^j)$ notes the curvature value of $(x_i, y_i)$ at its natural scale $\sigma^j$ (supposing that $C = \bigcup_{i=1}^L P_i$ where each curve part $P_i$ reveals a different structure; $(x_i, y_i) \in P_j$, and $\sigma^j$ is the natural scale to $P_j$). The set of dominant points is chosen between the local extremes of $K(\sigma^1, \ldots, \sigma^L)$.

To obtain the set of dominant points, once we have estimated the set $K_{ext}$ of local extremes, we cluster $K_{ext}$ using the dissimilarity measure defined as

$$M(i) = \min_{j < k} [d_{jk}] \quad j, k = 1, \ldots, i$$

(11)

where $i$ is the hierarchical level, which is the same as the number of clusters at the level; $d_{jk}$ is the distance between each cluster pair, which is defined as:

$$d_{jk} = \sqrt{\frac{n_j \times n_k}{n_j + n_k} \times |m_j - m_k|}, \quad j, k = 1, \ldots, i$$

(12)
where \( n_j \) and \( n_k \) are the number of samples in cluster \( j \) and \( k \) respectively; \( m_j \) and \( m_k \) are the mean values of the clusters.

In the clustering process, the set \( K_{ext} = \{k_1, \ldots, k_c\} \) is initially divided into a number of nonoverlapping classes \( K^1_{ext}, \ldots, K^c_{ext} \). In this initial partition, each extreme value \( k_i \) in \( K_{ext} \), is assigned to a different class: \( k_i \in K^i_{ext} \). A simple clustering is done based on the distance measure \( d_{jk} \) defined as in (12). First the minimum distance between any two clusters is found, and the two clusters with the minimum distance are grouped (to define a new partition) only if this grouping improves the dissimilarity between clusters. The dissimilarity is measured as

\[
D(i) = \frac{1}{i(i-1)} \sum_{j,k=1, \ldots, i \atop j \neq k} d_{jk} \tag{13}
\]

where \( i \) is the number of clusters. This dissimilarity measure is used to propose a stopping rule for assuring an adequate number of natural groupings (following the previous idea). Therefore, the clustering process requires neither the number of clusters in the data nor the minimum level of homogeneity within a cluster.

Let \( K^1_{ext}, \ldots, K^M_{ext} \) be the final partition of the set \( K_{ext} \), where it is not restrictive to suppose that the mean of the absolute value of curvature in \( K^i_{ext} \) is less than or equal to the mean of the absolute value of curvature in \( K^{i+1}_{ext} \), for all \( i \in \{1, \ldots, M-1\} \). Let \( C^1_{ext}, \ldots, C^M_{ext} \) be \( M \) subsets of the digitized contour \( C \), where a contour point \((x_i, y_i)\) is in \( C^j_{ext} \) only if \( k_i(\sigma^i) \) is in \( K^j_{ext} \); with \( k_i(\sigma^i) \) being the curvature value of \((x_i, y_i)\) at its natural scale \( \sigma^i \).

Let \( E(i) \) be defined as

\[
E(i) = \sum_{j=1}^{n} \epsilon_j^2 \tag{14}
\]

with \( \epsilon_j^2 \) being the distance of the point \((x_i, y_i)\), in the contour \( C \), from its approximating line segment in the polygon made joining the points in the subset \( C - \bigcup_{i=i+1}^{M} C^i_{ext} \). Thus, \( E(i) \) measure the global error between the digital curve \( C \) and the polygon made joining the successive points in the set \( C - \bigcup_{i=i+1}^{M} C^i_{ext} \). The method to obtain the dominant points set was based in the next idea: if the value \( i^* \) verifies the maximum decreasing of \( E(i) \) is
reached to $E(i^*)$, and therefore

$$E(i^* - 1) - E(i^*) = \max_{i=2,\ldots,M} \{ E(i - 1) - E(i) \}$$

we should expect that the subset of points $C_{ext}^{i^*}$ (and as a consequence the subsets of points $C_{ext}^{i^*+1}, \ldots, C_{ext}^{M}$) contains the most significant information between all the subsets $C_{ext}^i$, with $i = 1, 2, \ldots, M$, to build a polygonal approximation of the digitized contour $C$.

This idea is used to propose a reliable rule to determine the set of dominant points: the set of dominant points $D$ is defined as

$$D = C_{ext}^{i^*} \cup \cdots \cup C_{ext}^{M},$$

if

$$E(i^* - 1) - E(i^*) = \max_{i=2,\ldots,M} \{ E(i - 1) - E(i) \}$$

with $E(i)$ being defined as in (14).

4 The pattern recognition method

In this section, we describe a process for shift, rotation, and scale invariant pattern recognition using the shapes extracted from an image. Our method produces a feature-based pattern recognition, where the features used consist of polygonal approximations for the two-dimensional contours (each described at its scale vector) which are extracted from the shapes.

A variety of methods for extracting contours from an image have been proposed by different authors Gupta[9], Nagy[15], Nevatia[16], and Rosenfeld[20][21]. Extracted contours are broken and separated at points where they intersect. Some of the broken contours are then rejoined if they meet a criterion of mathematical continuity. To accomplish this task, a process described in Shepherd[25] have been used. Once we have obtained a set of non-self-intersecting contours, these ones are encoded using their polygonal approximations, and may be directly compared with similarly encoded patterns stored in memory, allowing them to be recognized.

In what follows, we will use the turning function $\Theta_P(s)$ as a representation of the boundary of a polygon $P$. The function $\Theta_P(s)$ measures the angle of the counterclockwise
tangent at the reference point $O$ on $P$’s boundary. Thus $\Theta_P(O)$ is the angle $\nu$ that the
tangent at the reference point $O$ makes with some reference orientation associated with the
polygon (such as the x-axis). We assume that each polygon is rescaled so that the total
perimeter length is 1; hence $\Theta_P$ is a function from $[0, 1]$ to $\mathbb{R}$. More interesting properties
for $\Theta_P$ are described in Arkin\textsuperscript{[1]}.

An analysis of the usefulness of this representation to compute distance function for
comparing two polygons is made in Arkin\textsuperscript{[1]}. In a similar way, we compute a distance
function for comparing two polygons $P_1$ and $P_2$ by looking at natural notions of distances
between the turning functions $\Theta_{P_1}(s)$ and $\Theta_{P_2}(s)$. The degree to which $\Theta_{P_1}$ and $\Theta_{P_2}$ are
similar can be measured by taking the distance between the functions $\Theta_{P_1}(s)$ and $\Theta_{P_2}(s)$
according to the $L_2$ metric. Define the $L_2$ distance between $P_1$ and $P_2$ as

$$\delta_2(P_1, P_2) = \left( \int_0^1 \left| \Theta_{P_1}(s) - \Theta_{P_2}(s) \right|^2 ds \right)^{\frac{1}{2}}$$

But $\delta_2$ has some undesirable properties: it is sensitive to both rotation of polygons and choice
of reference point on the boundary of $P_1$ (or $P_2$). Since rotation and choice of reference
point are arbitrary, it makes more sense to consider the distance to be the minimum over
all such choices. If we shift the reference point $O$ along $P_1$’s boundary by an amount $t$, then
the new turning function is given by $\Theta_{P_1}(s + t)$. If we rotate $P_1$ by angle $\theta$ then the new
function is given by $\Theta_{P_1}(s) + \theta$. Thus, we want to find the minimum over all such shifts $t$
and rotations $\theta$, which is equal to solving for

$$\left( \min_{\delta, \theta} \int_0^1 \left| \Theta_{P_1}(s + t) - \Theta_{P_2}(s) + \theta \right|^2 ds \right)^{\frac{1}{2}}$$

This distance can be used to make a method to compare polygons (and finally to recog-
nize patterns extracted from an image), which will be invariant under translation, rotation
and change of scale. The distance $d_2(\cdot, \cdot)$ is easy to compute, taking time $O(mn \log mn)$ to
compare an $m$ vertex polygon against an $n$ vertex polygon (Arkin\textsuperscript{[1]}).

5 Experimental results

The recognition process was applied to incoming images consisting of a set of non-overlapping
two-dimensional shapes (galaxies, countries, hands, leaves, etc). These images were digi-
tized using a scanner with a resolution of 300 DPI, and the boundaries were extracted using basic operations of mathematical morphology ($S_{erru}^{24}$).

In Figure 1, three 2-D contours are represented at their scale vectors. Any structure appearing on the curves is represented at its proper scale (the scale minimizing the normalized measure of the zeros of the curvature at a scale $\sigma$).

In Figure 2, we show four polygonal approximations obtained by joining the successive dominant points which were detected using the scale-vector based dominant point detection algorithm presented here.


To show to what extent the proposed method is better than (a) a weakened (e.g. single-scale) version of our approach, and (b) one standard method of boundary pattern analysis via $(s, k)$ plot, with $k$ being the digital curvature, the performance of the scale-vector scheme has been evaluated for a number of real pattern sets showing the differences with both (a) and (b).

In the single-scale version of the proposed method, to find out one scale for the entire boundary, the scale minimizing (9), with $C_i$ being equal to $C$, is chosen. The results obtained show that if the boundary is processed at such one scale then some significant detail from the boundary’s shape can be removed by over-smoothing. On the other hand, the standard method (b) uses the graph of the curvature values computed from the digital boundary $C$. As described in the subsection 2.1., the curvature value at each point has been estimated. The 1-D graph $(s, k)$ so obtained for the test pattern is matched against the corresponding graph for each stored model for some particular object, and the model-based recognition is achieved by identifying the best match.

In the first experiment described here, three leaf boundaries were used to test the comparative performance of the proposed pattern recognition process with the other ones.
Polygonal approximations to the original boundaries are shown in Figure 5. The three classes of shapes in the test were complicated planar curves as well as being almost indistinguishable by eye. For each original boundary, test pattern sets composed of samples of different sizes were collected from the boundary in various rotational positions. The boundaries were rotated over the range from 0 to $2\pi$ radians, and the different sizes of a particular boundary varied from 0.7 to 0.95 times the size of the original boundary. Twenty test samples per class were chosen for recognition. In Figure 6 the polygonal approximations to some of the test boundaries are shown. Our aim is to show that our approach is less sensitive to noise than both (a) and (b) as well as to study the robustness of the proposed method. Therefore, the test patterns were corrupted by normally distributed noise with zero mean and variance $\sigma^2$. Classification performance was measured by the number of correctly classified pattern set samples divided by the total number of samples tested. The entire process was repeated for $\sigma^2$ taking the values 1 and 2. Table 1 illustrates the comparative results of such a process showing that our technique is generally less sensitive to the noise. Moreover, from Table 1 we see that the scale-vector approach is relatively insensitive to the distributional shape of the noise considered here.

In the second experiment, three galaxy boundaries were utilized to examine the usefulness of the proposed methodology. Polygonal approximations to the original boundaries are shown in Figure 7. For each class, test data sets were collected as described above. In Figure 8 the polygonal approximations to some of the test boundaries are shown. Comparative results of this experiment are given in Table 2. As illustrated in Table 2, the proposed approach tends to be more resistant to the noise, and also we see from Table 2 that the use of the scale-vector method substantially improves the performance of the other ones considered here.

6 Conclusions

In this paper, a new process for shift, rotation, and scale invariant pattern recognition using shapes was introduced. It was shown how the shape representation results can be improved by using its two-dimensional contour which is described not at all different scales $\sigma$ but each curve part isolating a single structure at its proper scale. A polygonal approximation of a
planar curve was given by joining the successive dominant points detected on the contour represented at its scale vector. A scale-vector based dominant point detection algorithm was also presented. Model-based recognition was achieved by comparing the polygonal approximation to the contour extracted from shape A, which is stored as a model for some particular object, with the polygonal approximation to contour extracted from shape B, which is found to exist in an image. Because of the properties of the curve representation scheme and the method of comparing polygons itself, our vision system prove invariant to shape size as well as translational and rotational position. Experimental results showed that complicated planar curves could be recognized in high accuracy.

Acknowledgement

The authors would like to thank the anonymous referees for their many helpful comments.

References


Table 1.- The performance of the scale-vector recognition evaluated for three leaf boundaries showing the differences with both (a) a weakened (e.g. single-scale) version of our approach, and (b) one standard method of boundary pattern analysis via $(s, k)$ plot. In this table, the performance figures represent percentage of correct classification.

Table 2.- The performance of the scale-vector recognition evaluated for three galaxy boundaries showing the differences with both (a) a weakened (e.g. single-scale) version of our approach, and (b) one standard method of boundary pattern analysis via $(s, k)$ plot. In this table, the performance figures represent percentage of correct classification.

Figure 1.- (A), (B), (C) show some contours which are respectively described at their scale vectors in figures (D), (E), (F).

Figure 2.- Four polygonal approximations made by joining the successive dominant points which are extracted using the scale-vector based dominant point detection algorithm presented in Section 3.

Figure 3.- (A) shows an incoming pattern (a Spiral galaxy contour). (B) shows the contour described at its scale vector. (C) shows its polygonal approximation. (D) shows the best match for shape (A).

Figure 4.- (A) shows the incoming pattern (a Spiral galaxy contour). (B) shows the contour described at its scale vector. (C) shows its polygonal approximation. (D) shows the best match for shape (A).

Figure 5.- Polygonal approximations to the original boundaries in the first test.

Figure 6.- Polygonal approximations to some of the test shapes in the first experiment.

Figure 7.- Polygonal approximations to the original boundaries in the second test.

Figure 8.- Polygonal approximations to some of the test shapes in the second experiment.
<table>
<thead>
<tr>
<th>Test#1</th>
<th>noise $\sigma^2 = 1$</th>
<th>noise $\sigma^2 = 2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>“scale-vector” approach</td>
<td>96.6</td>
<td>93.3</td>
</tr>
<tr>
<td>“single-scale” version</td>
<td>78.3</td>
<td>70</td>
</tr>
<tr>
<td>standard method</td>
<td>83.3</td>
<td>75</td>
</tr>
<tr>
<td>Test#2</td>
<td>noise $\sigma^2 = 1$</td>
<td>noise $\sigma^2 = 2$</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>“scale-vector” approach</td>
<td>98.3</td>
<td>96.6</td>
</tr>
<tr>
<td>“single-scale” version</td>
<td>83.3</td>
<td>75</td>
</tr>
<tr>
<td>standard method</td>
<td>86.6</td>
<td>78.3</td>
</tr>
</tbody>
</table>
Figure 1
Figure 4
Figure 7