BINARY SHAPE RECOGNITION USING THE MORPHOLOGICAL SKELETON TRANSFORM

P. E. TRAHANIAS
Department of Electrical Engineering, University of Toronto, Toronto, Ontario, Canada M5S 1A4

(Received 1 July 1991; in revised form 5 March 1992; received for publication 16 March 1992)

Abstract—Binary image representation by its morphological skeleton transform has been proposed in the past as an information preserving representation. In this paper this skeleton representation is adopted as the start point for the development of a binary shape recognition scheme. A skeleton matching algorithm is presented that efficiently characterizes the similarity between two skeletons as a distance measure. This skeleton matching algorithm resembles the well known from speech processing elastic matching technique.

Morphological skeleton transform Binary images Shape representation Shape recognition

1. INTRODUCTION

The ultimate goal in many computer vision systems is the recognition of the shapes in the image and possibly the description (understanding) of the image.1-3 Shape recognition is usually accomplished by adopting a representation scheme that reduces a shape to a (small) number of shape descriptors and then classifying the shape according to the values of the descriptors. Alternatively, a shape can be represented by a set of shape primitives and possibly a set of primitive connection operators and then a structural recognition scheme can be used to perform the shape recognition task.1,2,3

A large number of shape descriptors has been reported so far.1-4 Pavlidis characterizes the shape descriptors as external or internal to distinguish between those which examine the boundary and those which examine the whole area of a shape. He also classifies them as information preserving and information nonpreserving depending on whether it is possible to reconstruct the original shape from the shape descriptor. From the pattern recognition point of view the above classifications are irrelevant as far as correct recognition is achieved. A representation scheme can potentially realize this goal if it carries all the information that is important for the class of shapes considered. In practice, it turns out that a representation scheme may be application dependent and this can be decided only through experimentation.

The skeleton of a shape has been proposed as an internal and information preserving descriptor which reduces a shape to an axial representation. The skeleton is actually a distance transform since it can be defined as the set of points whose distance from the nearest boundary is locally maximum.5 Among other techniques, the skeleton can be obtained by morphological set operations; this is usually referred to as the morphological skeleton transform (MST) to distinguish from the skeletons obtained using other approaches.6,7 It was pointed out by Maragos and Schafer that the MST unifies and generalizes previous approaches to skeletonization. A fast algorithm for skeleton decomposition has also been provided by the authors.

In this paper a scheme is presented for binary shape recognition which uses the MST representation. This scheme is based on a skeleton matching algorithm (SMA) which renders the similarity between two MSTs as a distance measure. The SMA resembles the well-known elastic matching technique that is widely used in the context of speech processing.6,9 Based on the distance measure, a given shape is classified as that shape from which its distance is minimum. Approaches to shape recognition based on mathematical morphology have also been proposed recently for binary4,10 and range images.11 These approaches are based mainly on shape decomposition into primitive components.

In what follows a brief introduction to morphological operations and the MST is first presented in Section 2 for the self completeness of the paper. The SMA is then described in Section 3 and the recognition scheme is presented in Section 4. Results of the application of the proposed SMA are described in Section 5 and the paper concludes with a brief discussion in Section 6.

2. THE MORPHOLOGICAL SKELETON TRANSFORM

2.1. Morphological operations

The fundamental morphological operations, which form the basis of most morphological transformations, are erosion and dilation.5,7,13 Erosion and dilation are initially defined for sets; however, their definitions can be extended to functions via the Umbra transformation.14 Since this paper deals with binary images (sets) only, the definitions of these operations will be given for this case only. The reader interested in function morphological operations may see references (7, 13).
Erosion $\ominus$ is a shrinking operation and dilation $\oplus$ is an expanding operation. For any binary shape $X$

$$X \ominus B = \bigcap_{b \in B} X - b = \{z : (B + z) \subseteq X\}$$

and

$$X \oplus B = \bigcup_{b \in B} X + b = \{x + b : x \in X \text{ and } b \in B\}$$

where $B$ represents a simple shape acting as a probe. It is called structuring element in morphological terminology. The output of an erosion is the set of translation points such that the translated structuring element is contained in the input set $X$. Similarly, the output of a dilation is the set of translated points such that the translate of the reflected structuring element $B = \{-b : b \in B\}$ has a non-empty intersection with $X$.

Two additional fundamental operations can be defined as combinations of erosion and dilation. Opening $\odot$ is an erosion followed by a dilation and closing $\bullet$ is a dilation followed by an erosion.

$$X \odot B = (X \ominus B) \oplus B$$

and

$$X \bullet B = (X \oplus B) \ominus B.$$ 

2.2. MST representation

The MST of a binary shape can be obtained by successive erosions and openings of the shape by a structuring element $B$. The basic algorithm has continuous and discrete versions. In the continuous case the MST of a binary shape $X$ is calculated by

$$SK(X) = \bigcup_{r > 0} S_r(X) = \bigcup_{r > 0} [(X \ominus rB) \setminus (X \ominus rB)_d]$$

where $\setminus$ denotes set difference, $rB$ an open disk of radius $r$ and $rB$ a closed disk of infinitesimally small radius $dr$.

For the discrete case the MST of a digital binary shape $X$ can be obtained by a similar algorithm

$$SK(X) = \bigcup_{n = 0}^{N} S_n(X) = \bigcup_{n = 0}^{N} [(X \ominus nB) \setminus (X \ominus nB)_d]$$

where $N = \max \{n : (X \ominus nB) \neq \emptyset\}$.

The sets $S_n n = 0, 1, \ldots, N$, are referred to as the skeleton components of $X$ with respect to $B$. From the sets $S_n$ $X$ can be reconstructed exactly as

$$X \odot nB = \bigcup_{k = 0}^{N} S_k \odot nB, \quad 0 \leq k \leq N.$$
Fig. 2. MST examples of binary shapes with respect to the "RHOMBUS" structuring element.
Algorithm, SMA
Input. $S_1, S_2$ are $S_1$ and $S_2$ stand for the two MSTs to be matched, $S_1 = SKF(X_1), S_2 = SKF(X_2)$ */
Output. $D$ /* A distance measure that expresses the similarity between $S_1$ and $S_2$ */

Method.
Step 1: (Initialization)
Set $D = 0$;
$n_1 =$ number of pixels in $S_1$;
$n_2 =$ number of pixels in $S_2$;
let $n_1 \geq n_2$, otherwise change the roles of $S_1$ and $S_2$

Step 2: for each pixel $(x,y) \in S_1$ do
begin
find its nearest pixel $(x',y') \in S_2$;
/* nearest is meant in the Euclidean sense */
compute $d = W[(x,y),(x',y')] R[(x,y),(x',y')]$;
/* $d$ represents the cost for matching pixel $(x,y)$ with pixel $(x',y')$ */
$W$ stands for weighting factor and is computed as
$W[(x,y),(x',y')] = |SKF(x,y) - SKF(x',y')| + 1$
$R$ stands for length of the straight line that connects the grid locations $(x,y)$ and $(x',y')$ /*
increase $D$ by $d$;
mark $(x',y')$ as visited;
end;

Step 3: for each unvisited pixel $(x',y') \in S_2$ do
begin
find its nearest pixel $(x,y) \in S_1$;
compute $d = W[(x',y'),(x,y)] R[(x',y'),(x,y)]$;
increase $D$ by $d$;
end;

Step 4: stop.

A few comments on this algorithm follow. The
computed distance between two MSTs by the SMA is not the minimum one since, although the expression \( R[(x,y), (x',y')] \) in the computation of \( d \) may or may not be minimum and hence the algorithm may reach a suboptimal solution. However, the choice of this "local" cost function is justified since (a) it yields a computationally cheap algorithm since the search for \((x', y')\) stops after locating the nearest pixel (usually after a few iterations). In the case that the global minimum would be searched for, an exhaustive search of all the pixels in \( S_2 \) is required and the risk exists in matching pixels that are located "far away" from \((x, y)\) in the image plane as opposed to the elastic matching which usually starts from a similarity of parts examination and then based on the overall geometry (spatial relations) the final decision is made.

By matching each pixel in the first MST with its closest in the second MST and accumulating the costs associated with each match into the global distance measure, the SMA actually performs a local non-linear alignment of the two MSTs in both directions. The cost accumulation in the global distance measure provides a means for determining the overall similarity or dissimilarity of two shapes. In other words, the distance measure is the final criterion for judging two shapes as similar or not. The use of the weighting factor \( W[(x,y),(x',y')] \) guarantees that each pair of pixels being matched has a large contribution to the whole distance measure if they represent different parts of the two shapes whereas it has a small contribution if they represent similar parts of the two shapes.

The computational complexity of this algorithm is difficult to compute analytically since the number of iterations needed to find the pixel \((x', y')\) that is the nearest to \((x, y)\) in \( S_2 \) is not known in advance. Similarly, not known in advance is the number of unvisited pixels from \( S_1 \) and hence the number of iterations in step 3. In the worst case the whole \( N \times N \) image grid would have to be searched in order to find the nearest pixel to a given pixel, and thus the worst case computational complexity of step 2 is \( O(n^2) \) and if we consider that \( n_1 \) pixels from \( S_1 \) have been visited in step 2 (in the worst case \( n_1 = 1 \)) then the computational complexity of step 3 is \( O((n_2 - n_1)n^2) \). Since \( n_1 \) and \( n_2 \) (number of pixels in \( S_1 \) and \( S_2 \), respectively) are usually in the order of \( N \), the above analysis results in a worst case complexity \( O(N^3) \). In practice, however, only a small number of iterations is needed in the search for a nearest pixel which, for all practical purposes, can be taken as a constant \( k \). Thus steps 2 and 3 have complexities \( O(nk) \) and \( O((n_2 - n_1)k) \), respectively, and taking as \( n = \max\{n_1, n_2 - n_1\} \) we conclude that in practice the complexity of the whole algorithm is \( O(n) \) which is roughly equal to \( O(N) \). The assumption of very small \( k \) was tested in practice and was found to be valid.

4. SHAPE RECOGNITION BASED ON SKELETON MATCHING

A shape recognition scheme is presented in Fig. 3 which is based on the SMA described in the previous section. The distance of the shape being examined from each of a set of reference shapes is computed by the SMA and the shape is classified as that reference shape from which its distance is a minimum. The reference shapes are not stored in their initial form; instead their MSTs are stored and this provides for large computational savings.

Since the MST representation is not invariant under scale and rotation, a normalization process is applied before the MST computation concerning scale and rotation. Translation normalization is also applied so that the shapes are placed in a predetermined position in the image plane. The shape normalization ensures representation invariance (and consequently recognition invariance) under rotations and scalings. These tasks are performed as follows.

**Translation normalization.** The axial system is moved so that the origin coincides with the center of mass of the shape. That is, if \((\bar{x}, \bar{y})\) are the coordinates of the center of mass and \(f(x,y)\) represents the initial binary shape, the translated function \(f_t(x,y)\) is given as \(f_t(x,y) = f(x+\bar{x}, y+\bar{y})\).

**Scale normalization.** The shape is enlarged or reduced so that the number of pixels \( n_2 \) in it becomes a predefined constant \( \beta \). This can be accomplished by transforming the function \(f(x,y)\) to \(f_n(x,y) = f(x/x', y/y')\), where \(x' = \sqrt{\beta n_2} \frac{1}{n_1}\). Both translation and scale normalization can be performed in one step by transforming \(f(x,y)\) to \(f_n(x,y)\), where \(f_n(x,y) = f(x + \bar{x}, y + \bar{y})\).

![Binary Shape Recognition Scheme](image_url)
Rotation normalization. The shape is rotated by an angle $\phi$ so that the angle between the major axis of the shape and the horizontal direction takes a predefined value $\psi$. Rotation is performed by computing the new location $(x', y')$ of each pixel $(x, y)$ as follows:

$$
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} =
\begin{bmatrix}
  \cos \phi & \sin \phi \\
  -\sin \phi & \cos \phi
\end{bmatrix}
\begin{bmatrix}
  x \\
  y
\end{bmatrix}
$$

The initial angle $\theta$ between the major axis and the horizontal direction is computed as

$$
\theta = \frac{1}{2} \arctan \left( \frac{2\mu_{11}}{\mu_{20} - \mu_{02}} \right)
$$

where $\mu_{pq}$ denotes the $(p + q)$th order central moment and is computed as

$$
\mu_{pq} = \sum_{x} \sum_{y} (x - \bar{x})^p (y - \bar{y})^q \cdot f(x, y).
$$

5. EXPERIMENTAL RESULTS

The proposed SMA has been implemented in the C language and some results concerning its application are presented here. For illustrative purposes, the detailed matching process is presented in Fig. 4 for two simple binary shapes $X_1 = '+'$ and $X_2 = 'H$'. In Fig. 4(a) the two binary shapes are shown and their MSTs, with respect to the "RHOMBUS" structuring element, are presented in Fig. 4(b). These shapes have intentionally been scaled using a small value for the constant $\beta (\beta = 125)$ in order to keep the number of pixels in the MSTs small and simplify the presentation of the matching process. The information contained in the skeleton functions is shown in Fig. 4(c) where for each pixel $(x, y) \in \text{MST}_i, i = 1, 2$, the value of the skeleton function $\text{SKF}(X_i)(x, y), i = 1, 2$, is plotted. In Fig. 4(d) the matching of the two skeletons is shown. In this figure the two MSTs are superimposed. For purposes...
Fig. 5. Set of binary images used as test set. The MSTs are superimposed on the shapes. From left to right and top to bottom: " + ", "H", ■", "rectangle", "ellipse", "rhombus", "X", "T", "triangle", "N", "E", "O".
Fig. 5. (Continued.)
Binary shape recognition using the morphological skeleton transform

Table 1. Confusion matrix for the 12 shapes of Fig. 5

<table>
<thead>
<tr>
<th>Shape</th>
<th>+</th>
<th>H</th>
<th>rectangle</th>
<th>ellipse</th>
<th>rhombus</th>
<th>X</th>
<th>T</th>
<th>triangle</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>0</td>
<td>175</td>
<td>107</td>
<td>137</td>
<td>265</td>
<td>301</td>
<td>184</td>
<td>126</td>
</tr>
<tr>
<td>H</td>
<td>175</td>
<td>0</td>
<td>87</td>
<td>289</td>
<td>540</td>
<td>639</td>
<td>148</td>
<td>187</td>
</tr>
<tr>
<td>rectangle</td>
<td>107</td>
<td>87</td>
<td>0</td>
<td>80</td>
<td>185</td>
<td>192</td>
<td>135</td>
<td>90</td>
</tr>
<tr>
<td>ellipse</td>
<td>137</td>
<td>289</td>
<td>0</td>
<td>26</td>
<td>70</td>
<td>98</td>
<td>139</td>
<td>345</td>
</tr>
<tr>
<td>rhombus</td>
<td>265</td>
<td>540</td>
<td>185</td>
<td>26</td>
<td>0</td>
<td>22</td>
<td>170</td>
<td>268</td>
</tr>
<tr>
<td>X</td>
<td>184</td>
<td>148</td>
<td>135</td>
<td>98</td>
<td>170</td>
<td>235</td>
<td>0</td>
<td>151</td>
</tr>
<tr>
<td>T</td>
<td>126</td>
<td>187</td>
<td>90</td>
<td>139</td>
<td>268</td>
<td>353</td>
<td>151</td>
<td>361</td>
</tr>
<tr>
<td>triangle</td>
<td>265</td>
<td>502</td>
<td>135</td>
<td>345</td>
<td>513</td>
<td>583</td>
<td>619</td>
<td>151</td>
</tr>
<tr>
<td>⊗</td>
<td>191</td>
<td>107</td>
<td>187</td>
<td>282</td>
<td>433</td>
<td>509</td>
<td>130</td>
<td>226</td>
</tr>
<tr>
<td>⊠</td>
<td>168</td>
<td>91</td>
<td>256</td>
<td>369</td>
<td>782</td>
<td>782</td>
<td>151</td>
<td>130</td>
</tr>
<tr>
<td>⊘</td>
<td>146</td>
<td>151</td>
<td>276</td>
<td>294</td>
<td>559</td>
<td>622</td>
<td>190</td>
<td>219</td>
</tr>
</tbody>
</table>

The set of geometrical shapes shown in Fig. 5 has been used as a test set for the SMA. This set consists of 12 shapes which are shown after being normalized. For the scale normalization a value of $\beta = 500$ has been selected. In this figure, the MSTs of the shapes are also shown superimposed on the actual shapes. The distance measures computed by the SMA for the binary shapes of Fig. 5 are given in the form of a confusion matrix in Table 1. As can be verified, similar shapes result in much smaller distance measures than other shapes, which coincides with our perception of shape similarity. Shape "1--1" for example, appears to be very similar to shape "H" and also similar to shape "□". Similarly, shapes "rectangle", "ellipse" and "rhombus" appear as similar.

A second experiment has been conducted concerning the recognition of the shapes shown in Fig. 5 after being corrupted by a "boundary deformation" procedure. This procedure produces a noisy version of a binary shape in two steps: (a) for each contour pixel of the initial shape a pixel is randomly selected from the set consisting of the contour pixel and the background pixels neighboring it, and (b) the value of the selected pixel is changed with a probability $\gamma$. Higher deformations can be produced by increasing $\gamma$ or by repeating this process $\delta$ times for a given $\gamma$. This process affects the shape representation by skewing the MST and by introducing spurious pixels in the MST at or near the shape contour. Sample noisy shapes, along with their MSTs, are shown in Fig. 6 for $\delta = 3$ and $\delta = 4$, and $\gamma = 10\%$. This value of $\gamma$ has been kept constant in this experiment whereas $\delta$ has been varied from 1 to 10. For each value of $\delta$ ten deformed versions of each shape have been produced. In summary, a set of 1200

Table 2. Recognition errors for the shapes of Fig. 5 deformed at various levels of $\delta$

<table>
<thead>
<tr>
<th>Shape</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>H</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rectangle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ellipse</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rhombus</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>X</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>T</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>triangle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>⊗</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>⊠</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>⊘</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Fig. 6. Noisy shapes produced with the "boundary deformation" procedure: (a) $\lambda = 3$; (b) $\lambda = 4$. 
deformed shapes has been produced (100 for each shape, or 120 for each value of \( \delta \)). This set has been used as a test set, and the set of initial shapes has been used as the reference library in the experiment (12-class problem). The recognition errors for the various values of \( \delta \) and for each shape are given in Table 2. As can be verified from this table, no errors were made for (a) \( \delta = 1, 2, 3 \) and all the classes of shapes, and (b) all \( \delta \) values and eight out of the twelve shape classes. The majority of errors were observed for the "ellipse" that was misrecognized mostly as "rectangle". The total number of errors was 68 which accounts for a recognition rate of 94.33\% for this specific set of noisy shapes.

- It can be applied to any type of binary shape without making any assumptions about its outer form or geometrical distribution.

Extensive testing of the SMA is however still needed and is planned for real shapes, e.g. machine parts, characters, etc. Other future work on this subject includes the extension of the SMA to graytone images using the morphological skeleton for graytone images which is defined as the pointwise sum of the skeleton components \( S_n \):

\[
S_n = \left( f \circ \mathbb{g} \right) - \left( \left( f \circ \mathbb{g} \right) \circ \mathbb{g} \right), \quad 0 \leq n \leq N \quad (14)
\]

where \( f \) represents the graytone image function, \( N = \max \{ n : f \circ \mathbb{g} \neq \mathbb{g} \} \) and \( g \) the structuring element which can be a binary or a graytone function.

Acknowledgements—The reviewers' suggestions and comments were very helpful and greatly appreciated by the author.

6. DISCUSSION

A binary shape recognition scheme has been presented in this paper. This scheme adopts the MST as the shape representation formalism. In the heart of this shape recognition scheme lies the proposed SMA which is responsible for the decision regarding the similarity between two shapes. The SMA attempts to perform a local skeleton matching that is weighted according to the significance (skeletal function) of the skeleton points in each representation. The SMA is based on similar concepts as the elastic matching technique but in order to achieve computational efficiency it reaches a suboptimal solution instead of the globally optimal one.

Shape normalization is performed before the application of the SMA in order to ensure invariance in shape representation (and recognition), under rotations and scalings. This process is not necessarily optimal with respect to the matching performed and cost minimization with respect to normalization could be applied. Such a solution has not been employed, however, due to the increase in the computational cost associated with it. This is due to the fact that cost minimization would require examination of many possible normalizations and internal skeleton point correspondences that would make the algorithm prohibitively slow.

The SMA has been tested in practice using a set of geometrical shapes and its performance has been found to be very promising. Its robustness when applied to noisy data has been demonstrated using a set of deformed shapes at various levels. Besides being accurate, the proposed approach possesses some very important characteristics:

- It is simple and intuitively appealing since it expresses the similarity between two shapes based on the alignment of their MSTs which can be thought of as the thinned sketches of the shapes.
- It is computationally very efficient since both the MST and the SMA operate in \( O(N) \) time.

REFERENCES

About the Author — PANAGIOTIS TRAHANIAS received the B.S. in physics from the University of Athens, Greece, in 1985 and the Ph.D. in computer science from the National Technical University of Athens, Greece, in 1988. From 1985 to 1989 he served as a research assistant at the Institute of Informatics & Telecommunications, National Center for Scientific Research "Demokritos", Greece, and from 1990 to 1991 he worked at the same institute as a postdoctoral research associate. Since October 1991 he has been with the Department of Electrical Engineering, University of Toronto, Toronto, Ontario, where he holds a postdoctoral research position. His research interests include pattern recognition, signal/image processing and computer vision and image analysis. Dr. Trahanias is a member of the IEEE Society.