AUTOMATIC SELECTION OF STRUCTURING ELEMENT FOR BENGALI NUMERAL RECOGNITION

BHABATOSH CHANDA

Electronics & Communication Sciences Unit, Indian Statistical Institute,
203 B.T. Road, Kolkata 700108, India
chanda@isical.ac.in.

BIDYUT B. CHAUDHURI

Computer Vision & Pattern Recognition Unit, Indian Statistical Institute,
203 B.T. Road, Kolkata 700108, India
bbc@isical.ac.in.

How to select a structuring element for a given task is one of the most frequently asked questions in morphology. The present work tries to find a solution for a restricted class of problems, in the domain of shape classification. In this work an algorithm that extracts distinctive structure of each of a given set of objects, which can be used as the structuring elements for object classification system employing the hit-and-miss transformation, is proposed. The proposed algorithm is based on a new measure of local shape property. The method is used to develop a system for Bengali numeral recognition.

Keywords: Binary object; mathematical morphology; hit-and-miss; structuring element; shape classification; distance transforms.

1. Introduction

Mathematical morphology is becoming a more and more popular tool for dealing with the shapes of objects in an image.[4,5,12] The main difference between a mathematical morphological and traditional image processing techniques is that the former treats image as an ensemble of sets rather than a signal. The language of mathematical morphology is that of set theory and the operations are defined in terms of the interaction between two sets: an object and a structuring element. Morphological operations can simplify image data preserving by their essential shape characteristics and by eliminating irrelevancies. It is natural that mathematical morphology has an important role to play in machine vision.[7,9,13,14,16] Automatic identification and classification of objects in an image are major applications and many interesting algorithms have been reported.[1,6,10,15] A strong point in favor of morphology is that each step of morphological algorithms is mathematically well founded because given an input image and structuring element, the output of a morphological operation may be determined analytically. So, selection of operations and their sequence to accomplish a task are less heuristic in case of morphological approaches.

However, one of the main disadvantages of this class of algorithms is that the procedure for selecting a structuring element is completely *ad hoc* in nature. In other words, the algorithm designer selects a suitable structuring element depending on types of objects and application in hand using his intuition and experience. Therefore one of the most sought after question in designing morphological algorithms is whether we can automatically select appropriate structuring elements for a given problem. Not much work can be found in the literature in this direction. In Ref. 2, Chanda and Bhattacharya tried to find structuring element automatically for object recognition through erosion. Though the system worked well, the selection procedure was very cumbersome and required lots of search and comparison. The present work may be considered as a much-improved version of that early work. Another work attempted to select automatically optimal structuring element for shape description and matching based on morphological signature transform by using genetic algorithms. However, its success depends on the design of optimization criteria and the convergence of the algorithm.[8]

Providing a general solution to this morphological structuring element selection problem is beyond the scope of a single work. So we restrict ourselves within the problem of object classification. An important and widely used object detection/classification/recognition is template matching.[3,11] In case of a gray-scale image correlation with a template at each position is measured. In case of a binary image, match between the image and template as well as the match between complement (or reverse) image and the complemented template are measured at each position. An equivalent operation to this is hit-and-miss transform in mathematical morphology. Here, we have proposed an algorithm to extract a distinctive shape characteristic of an object with respect to a given set of objects. This leads to automatic selection of structuring elements for a major class of applications, namely object classification by hit-and-miss operation. The structuring elements selected through this proposed algorithm are used to develop a system for Bengali numeral recognition.

This paper is organized as follows. Section 2 presents a formal definition of the problem. A plausible solution is given is Sec. 3. A proposed algorithm is also described in this section. In Sec. 4, a brief description of implementation and the experimental results are given. Finally, concluding remarks are provided in Sec. 5.

2. A Formal Definition of the Problem

In a continuous domain R^2 an image is defined as a nonnegative function f where $f(x, y)$ is the value of the function at the point (x, y). A digital image is defined by a finite-value function over a discrete domain Z^2. Let us assume a digital image domain $I \subset Z^2$ is

$$I = \{(r, c) \mid r = 0, 1, \ldots, M-1; c = 0, 1, \ldots, N-1\}$$

obtained by sampling of step size h along two orthogonal directions. The points of digital images are often called pixels and are denoted by p or the coordinate (r, c). Here, we only consider binary images, i.e. $f(r, c) \in \{0, 1\}$. A pixel (r, c) in I is defined as a foreground pixel or object pixel if its value is 1, and as a background pixel if its value is 0. Therefore, a digital object A is a connected set of pixels of I. Here, we will use the conventional definitions of all the relevant terms like neighborhood, connectedness etc.

Definition. Let p and q be two discrete pixels in Z^2 with coordinates (r_p, c_p) and (r_q, c_q), respectively, then the *distance* between p and q may be defined as

$$d_4(p, q) = |r_p - r_q| + |c_p - c_q|$$
$$d_8(p, q) = \max\{|r_p - r_q|, |c_p - c_q|\}\,.$$

So, in the discrete domain, a circular region defined using d_4 distance metric looks like a diamond, and that using d_8 looks like a square. Unless explicitly specified, we assume objects as 8-connected component and the background as 4-connected.

Definition. Let us define a *threshold function* $t(\nu, \theta)$, where ν and θ are real numbers, as

$$t(\nu, \theta) = \begin{cases} 1 & \text{if } \nu \geq \theta \\ 0 & \text{otherwise} \end{cases}\,. \tag{1}$$

Thus, for any gray-level image f, the threshold function t operated on f defined as

$$t(f, \theta) = t(f(r, c), \theta)$$

produces a binary image.

The primary morphological operations are dilation and erosion. From these two, other important operations, namely opening, closing and hit-and-miss can be constructed. Dilation enlarges an object by adding pixels (layer) around the boundary of object. Erosion shrinks an object by removing pixels along the boundary, but retaining the internal structure of the object. Open operation smoothes the contour, breaks narrow isthmuses, and eliminates small islands and sharp corners. Close operation smoothens the contours, fuses narrow breaks and long thin gulfs and eliminates small holes. Hit-and-miss transform detects whether or not a particular shape feature is present in an object of image. We use standard definitions for all these operations and do not present them except two, namely erosion and hit-and-miss, as follows.

Definition. The *erosion* of a set A by a structuring element B denoted by $A\Theta B$, is defined by

$$A\Theta B = \{p \in Z^2 \mid B_p \subseteq A\}\,.$$

Definition. The *hit-and-miss* of a set A by a pair of structuring element (B, C) denoted by $A \otimes (B, C)$, is defined by

$$A \otimes (B, C) = (A \Theta B) \cap (A^c \Theta C) .$$

It should be maintained that $B \cap C = \Phi$, otherwise $A \otimes (B, C)$ will always be null.

2.1. *Problem definition*

In most of the morphological image processing and analysis algorithms, the structuring elements are selected in an *ad hoc* basis depending on the problem and the class of images. In this work, on the other hand, we try to devise an algorithm that can select appropriate structuring element automatically for a class of problems, namely 2D object recognition. The problem may be formally presented as follows.

Given the m distinct class C_k, $k = 0, 1, 2, \ldots, m$ of objects where A_{ki} is an object of class k, i.e. $A_{ki} \in C_k$, our problem is to find a set of m minimal structuring element pairs $\{(B_k^0, B_k^c) \mid k = 0, 1, 2, \ldots, m\}$ such that

$$A_{ji} \otimes (B_k^0, B_k^c) = \begin{cases} \neq \Phi & \text{for } j = k \\ = \Phi & \text{otherwise} \end{cases} \tag{2}$$

for all i. By the term "minimal structuring element" we mean the structuring element that has the smallest domain of definition. The minimality is important from a computational point of view. Suppose, an area of (or number of pixels in) the domain of definition of a structuring element B is denoted by $\partial(B)$ and the domain of definition of a pair of structuring elements (B_k^0, B_k^c) is the same as that of B_k, where $B_k = B_k^0 \cup B_k^c$. Since (B_k^0, B_k^c)s are extracted by analyzing the $A_{ki}s$ for all k and i, the solutions are not guaranteed to be unique. That means there may exist more than one pair of structuring elements for a class k that satisfy Eq. (2). Suppose they are (B_{k0}^0, B_{k0}^c), (B_{k1}^0, B_{k1}^c), (B_{k2}^0, B_{k2}^c), etc. Then the pair of minimal structuring elements is given by (B_k^0, B_k^c) if

$$\partial(B_k) = \min \partial(B_{ki}) . \tag{3}$$

For brevity, let each class contain only one object. We first describe the procedure of automatic selection of structuring element based on this assumption, and later we will generalize it to accommodate the real situation where a class contains more than one object. That means currently the object A_k completely describes the class C_k. Then Eq. (2) suggests that $B_k^0 \subseteq A_k$ and $B_k^c \subseteq A_k^c$ and also suggests that $B_k^0 \nsubseteq A_j$ and $B_k^c \nsubseteq A_j^c$ for $j \neq k$. Since B_k^0 and B_k^c may be selected from many different possibilities each of which is distinctive and satisfies aforementioned requirements, examining all these possibilities requires a mammoth amount of computation. So we restrict our search to the pair of structuring elements by assuming that B_k is a circular region of radius ρ.

Definition. A pair of minimal structuring elements (B_k^0, B_k^c) may be defined as a partition of a circular region B_k of radius ρ, if

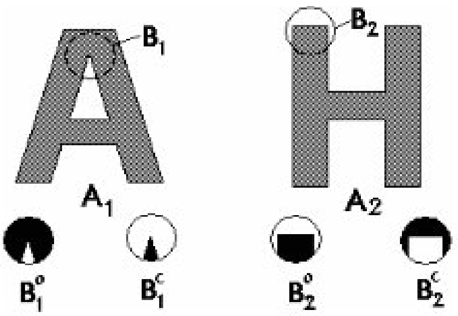

Fig. 1. An example of pairs of structuring elements for two objects A_1 and A_2. Circular region superposed on the objects are B_1 and B_2. Each of them is partitioned into gray-filled region and white region. Solid regions are elements of the pairs. Note that other distinctive pairs can also be extracted.

1. (B_k^0, B_k^c) satisfies Eq. (2).
2. $B_k = B_k^0 \cup B_k^c$ and $B_k^0 \cap B_k^c = \Phi$.
3. If there exist a series of circular regions B_ks of $\rho_1, \rho_2, \rho_3, \ldots$ that satisfy conditions 1 and 2, then $\rho = \min\{\rho_1, \rho_2, \rho_3, \ldots\}$.

Figure 1 gives an example of such a circular region, partition and corresponding structuring elements.

3. A Plausible Solution

An underlying idea of the proposed solution is to formulate a measure of local shape of an object over a circular region around each pixel of the image, and then extract the region that has a distinctive shape measure compared to other objects. Since the shape is defined by mutual arrangement of object pixels and the background pixels, we extend the concept of distance transform for this purpose. Distance transform is considered because it can preserve shape information. An example that shows distance transform can represent shape information is when an object skeleton can be straightway extracted from it. Let a binary image $f(r, c)$ denote the *indicator function* of a binary object or point set. Let us begin our discussion by defining distance transform of binary image as given below.

Definition. *Distance transform* maps a binary image matrix f to another matrix D where

$$D(r, c) = \min_{(k,l)}\{d_n((r, c), (k, l))\} \tag{4}$$

such that $f(k, l) = 0$. The value of n may be 4 or 8 depending on whether the object is 4-connected or 8-connected, respectively. So, each element of $D(r, c)$ represents the distance of a pixel(r, c) from the background. There are different ways of finding distance transform of an image.[3,11] Here, we propose another way of computing it through convolution and threshold function.

Suppose $R_\rho(p)$ denotes a set of 1-pixels (coordinates relative to p) within a circular region $X_\rho(p)$ of radius ρ around the pixel p and $\#R_\rho(p)$ denotes the number of elements in $R_\rho(p)$ i.e.

$$R_\rho(p) = \{q - p \,|\, d_n(p,q) \leq \rho \text{ and value at } q = 1\}\,.$$

Then $\#R_1(p)$ is the number of 1-pixels within a circular region of radius 1 around p. Similarly, we can define $R^c_\rho(p)$, a set of 0-pixels within a circular region $X_\rho(p)$ of radius ρ around p, by letting $q = 0$. Now, we define surroundedness as follows.

Definition. The black or 1-*surroundedness* of p, denoted by $s_\rho(p)$ is defined as

$$\begin{aligned} s_1(p) &= \#R_\rho(p) \\ s_\rho(p) &= \sum_{q \in R_1(p)} s_{\rho-1}(q) \quad \text{for } \rho > 1\,. \end{aligned} \tag{5}$$

Hence, $s_\rho(p)$ reflects the number of 1-pixels around p at different distances, and thus gives an idea of distance of p from the background. For example, if all the pixels within a circular region of radius ρ around p be one then the values of $s_\rho(p)$ becomes α^ρ where $\alpha = 5$ or 9 depending on the 4- or 8-connectivity of the object. Since the pixels, which we are not interested in, have value 0, we may rewrite Eq. (5) as

$$\begin{aligned} s_1(p) &= \#R_\rho(p) \\ s_\rho(p) &= \sum_{q \in X_1(p)} s_{\rho-1}(q) \quad \text{for } \rho > 1\,. \end{aligned} \tag{6}$$

We can, similarly, define the white or 0-surroundedness of p, denoted by $s_\rho(p)$, by using $\#R^c_1(p)$. Thus the concept leads to computing the distance transform in the following way.

Let us first define two masks:

0	1	0
1	1	1
0	1	0

Therefore, the convolution operation f^*N_n puts the sum of all the n-neighbours' of (r,c) in f. We compute the distance transform matrix $D(r,c)$ iteratively as

$$
\begin{aligned}
S_0 &= f & D_0 &= t(S_0, \alpha^0) \\
S_1 &= S_0{}^* N_n & D_1 &= t(S_1, \alpha^1) + D_0 \\
S_2 &= S_1{}^* N_n & D_2 &= t(S_2, \alpha) + D_1 \\
&\;\;\vdots & &\;\;\vdots \\
S_l &= S_{1-1}{}^* N_n & D_l(r,c) &= t(S_l, \alpha^1) + D_{1-1}\,.
\end{aligned}
$$

Then the distance transform matrix D is same as D_1 if $D_l = D_{l+1}$. The value of α is 5 if $n = 4$ and 9 if $n = 8$. It should be noted that the element $s_\rho(r,c)$ of the surroundedness matrix S_ρ gives the black or 1-surroundedness around $p = (r,c)$ within a circular region of radius ρ [see Eq. (6)]. Secondly, $s_\rho(r,c)$ is monotonically nondecreasing with ρ. It should also be noted that if $R_\rho(p) \subset R_\rho(q)$ then $s_\rho(p) < s_\rho(q)$.

Proposition 1. *If* $R_\rho(p) = R_\rho(q)$ *then* $s_\rho(p) = s_\rho(q)$.

Proof. $R_\rho(p) = R_\rho(q)$ implies that the sets of pixels which are at a distance ρ or less are same for both p and q. Since $s_\rho(\cdot)$ is defined as a position-invariant linear transformation, we must have $s_\rho(p) = s_\rho(q)$. □

Proposition 1a. *If* $s_\rho(p) \neq s_\rho(q)$ *then* $R_\rho(p) \neq R_\rho(q)$.

Proof. Suppose not. That means $s_\rho(p) \neq s_\rho(q)$ implies $R_\rho(p) = R_\rho(q)$. But from Proposition 1, $R_\rho(p) = R_\rho(q)$ implies that $s_\rho(p) = s_\rho(q)$. Hence, the contradiction! □

So $s_p(p)$ gives an idea of concentration of 1-pixels around p within a circular neighborhood $X_\rho(p)$ of radius ρ, but it cannot reveal how these neighboring pixels are spatially distributed around p over that region. That means, though, $R_\rho(p) = R_\rho(q)$ implies $s_\rho(p) = s_\rho(q)$ and $s_\rho(p) \neq s_\rho(q)$ implies $R_\rho(p) \neq R_\rho(q)$ the reverse is not, in general, true. In other words, $s_\rho(p) = s_\rho(q)$ does not guarantee that $R_\rho(p) = R_\rho(q)$. One possible way to extract the information about spatial distribution of pixels is to assign the position variant weights to neighboring pixels before computing the surroundedness. Let us call this quantity *weighted surroundedness* and denote it by $w_\rho(p)$. So $w_\rho(p)$ may be defined as

$$
\begin{aligned}
w_1(p) &= (f^* W_n)(p) \\
w_\rho(p) &= (w_{\rho-1}{}^* W_n)(p) \quad \text{for } \rho > 1
\end{aligned}
\tag{7}
$$

where W_n is a weight matrix and n can be 4 or 8 as stated before. Examples of weight matrices for two cases are

$$W_4 = \begin{array}{|c|c|c|} \hline 0 & 2 & 0 \\ \hline 16 & 1 & 4 \\ \hline 0 & 8 & 0 \\ \hline \end{array} \quad \text{and} \quad W_8 = \begin{array}{|c|c|c|} \hline 32 & 2 & 64 \\ \hline 16 & 1 & 4 \\ \hline 256 & 8 & 128 \\ \hline \end{array}$$

We may define weight matrix in various other ways too. However, care should be taken so that weights at any position must not be equal to the sum of weights of other positions.

Proposition 2. *$R_\rho(p) = R_\rho(q)$ if and only if $w_\rho(p) = w_\rho(q)$.*

Proof. $R_\rho(p) = R_\rho(q)$ implies that the sets of pixels which are at a distance ρ or less are same for both p and q. Since $w_\rho(\cdot)$ is defined as a shift-invariant linear transformation, we must have $w_\rho(p) = w_\rho(q)$. On the other hand, since $w_\rho(\cdot)$ is obtained from convolving with a mask with different weights and different positions, if $R_\rho(p) \neq R_\rho(q)$, then $w_\rho(p) \neq w_\rho(q)$. □

Hence, $w_\rho(p)$ uniquely describes the spatial distribution of 1-pixels in a circular region $X_\rho(p)$ of radius ρ around p. We can, similarly, define $w_\rho(p)$ for the spatial distribution of 0-pixels in $X_\rho(p)$ by using $f(r,c)$, the indicator function of the complement of the object or the point set. Now, let us consider three different situations, considering the values of $w_\rho(p)$ and $w_\rho(q)$:

1. $w_\rho(p) = w_\rho(q)$: In this case according to Proposition 2, $R_\rho(p) = R_\rho(q)$. This implies that $R^c_\rho(p) = R^c_\rho(q)$ which results in $w_\rho(p) = w_\rho(q)$. That means local shape defined over a circular region $X_\rho(\cdot)$ at p and q are the same.
2. $w_\rho(p) > w_\rho(q)$: In this case $R_\rho(p)$ has either more pixels than that of $R_\rho(q)$ or it has at least one pixel at a position where the mask weight is higher than the positions occupied by the pixels of $R_\rho(q)$. In both situations, $R_\rho(p) \subseteq R_\rho(q)$. However, it may so happen that $R_\rho(p) \supset R_\rho(q)$ and in that case $R^c_\rho(p) \subseteq R^c_\rho(q)$. In both the situations we have $w_\rho(p) < w_\rho(q)$.
3. $w_\rho(p) < w_\rho(q)$: This case is exactly reverse of the previous one. So, we can either have $R_\rho(p) \subset R_\rho(q)$ or $R_\rho(p) \supseteq R_\rho(q)$. In the former situation $R^c_\rho(p) \supset R^c_\rho(q)$. However, in both cases, $w_\rho(p) > w_\rho(q)$.

Thus, the pair of numbers $(w_\rho(p), w_\rho(q))$ can be used as a local shape measure and the pair of sets $(R_\rho(p), R^c_\rho(q))$ represents the corresponding shape.

3.1. *Proposed algorithm*

Considering the given problem, for the object A_k the circular region $X_\rho(p)$ for some ρ is equivalent to the domain of definition of B_k. In that case, $R_\rho(p) = B^0_k$ and $R^c_\rho(p) = B^c_k$ if they satisfy Eq. (2). Accordingly, at any point p of A_k we denote the shape measure by the pair $(w_{k,\rho}(p), w_{k,\rho}(p))$. In the proposed algorithm (as shown in Fig. 2) we work with the binary images $f_k(r,c)$ and $f'_k(r,c)$ for $k = 0, 1, 2, \ldots, m-1$ which are indicator functions of A^0_k and A^c_k, respectively.

ALGORITHM-1:

Input: $f_k(r,c)$ and $f'_k(r,c)$ for all k.
Set $\rho = 0$.
Repeat {
    Increase ρ by 1
    For all k **do** {
        Compute $w_{k,\rho}(r,c)$ and $w'_{k,\rho}(r,c)$ at every (r,c).
        Retain only distinct pairs $(w_{k,\rho}(r,c), w'_{k,\rho}(r,c))$.
        Count the number $total(k)$ of these distinct pairs for each k.
        }
    For all k, j and $j \neq k$ **do** {
        If $(w_{k,\rho}(r,c), w'_{k,\rho}(r,c)) = (w_{j,\rho}(r,c), w'_{j,\rho}(r,c))$ for some k and j,
        Remove all of them from the list and reduce the count $total(\cdot)$
        }
    $\min\_total = \min_k\{total(k)\}$
    } **until** $(\min\_total = 0)$
Output: $(R_{k,\rho}(r,c), R^c_{k,\rho}(r,c))$ for every k as (B^0_k, B^c_k).
    Corresponding to any $(w_{k,\rho}(r,c), w'_{k,\rho}(r,c))$ remaining in the list.

Fig. 2. Algorithm for selecting structuring element in the special case when each class contains one object only.

The above discussion and Algorithm 1 ensure that (B^0_k, B^c_k) for $k = 0, 1, 2, \ldots, m-1$ can be used to recognize the class C_k when it contains only one object. However, when each class contains more than one object, the algorithm should work with a representative of each class. Suppose, A_{ki} is an object of class k, i.e. $A_{ki} \in C_k$. Then, to satisfy Eq. (2), we must ensure that

1. $B^0_k \subseteq A_{ki}$ for all i,
2. $B^c_k \subseteq A^c_{ki}$ for all i,
3. $B^0_k \nsubseteq A_{ji}$ for any j, $j \neq k$ and any i, and
4. $B^c_k \nsubseteq A^c_{ji}$ for any j, $j \neq k$ and any i.

These conditions suggest that the structuring element for each class should be determined separately, unlike in Algorithm 1 where all k pairs of structuring elements are determined in a single run. Thus, to determine the structuring element pair (required by hit-and-miss operation) for the class C_k we must take

$$A_k = \bigcap_i A_{ki} \quad \text{and} \quad A^c_k = \bigcap_i A^c_{ki}$$

and

$$A_j = \bigcup_i A_{ji} \quad \text{and} \quad A^c_j = \bigcup_i A^c_{ji} \text{ for } j \neq k\,.$$

ALGORITHM-2:

Input: $f_j(r,c)$ and $f'_j(r,c)$ for all j, and the class number k.
Set $\rho = 0$.
Repeat {
Increase ρ by 1.
For all j **do** {
Compute $w_{j,\rho}(r,c)$ and $w'_{j,\rho}(r,c)$ at every (r,c).
Retain only distinct pairs $(w_{j,\rho}(r,c), w'_{j,\rho}(r,c))$.
Count the number $total(j)$ of these distinct pairs for each j.
}
For all j and $j \neq k$ **do** {
If $w_{k,\rho}(r,c) \leq w_{j,\rho}(r,c)$ *and* $w'_{k,\rho}(r,c) \leq w'_{j,\rho}(r,c)$ *for some* j,
remove all of them from the list and reduce the count $total(\cdot)$.
}
$\min\_total = \min_j\{total(j)\}$
} **until** ($\min\_total = 0$)
Output: $(R_{k,\rho}(r,c), R^c_{k,\rho}(r,c))$ *as* (B^0_k, B^c_k)
corresponding to any $(w_{k,\rho}(r,c), w'_{k,\rho}(r,c))$ remaining in the list.

Fig. 3. Algorithm for selecting structuring element for the class k when each class contains more than one object.

As before $f_k(r,c)$ and $f'_k(r,c)$ are the indicator functions of A_k and A^c_k for all k defined above. This leads to Algorithm 2 (as shown in Fig. 3) which is a modified version of Algorithm 1 for the general case.

Hence, Algorithm 2 has to be executed for each of the object classes separately to obtain all the required structuring elements.

4. Experimental Results

We have implemented Algorithm 2 on a SGI machine with UNIX operating system. Structuring elements are selected automatically with an objective to build a recognition system for the Bengali numerals. That means we need 10 pairs of structuring elements (B^0_k, B^c_k), $k = 0, 1, 2, \ldots, 9$ such that the recognition system would declare an input object A as the numeral "j" if

$$\#(A \otimes (B^0_k, B^c_k)) = 0$$

for all k except j. If there exists no such j or more than one j we conclude that the system has failed to recognize.

It is a well-known fact that morphological structuring elements are neither rotation-invariant nor scale-invariant. Hence, to obtain an useful result, all the objects A_{ki}, should be brought to a fixed orientation and a fixed size. This may be done by rotating each object by such an angle so that their, say, principal axis

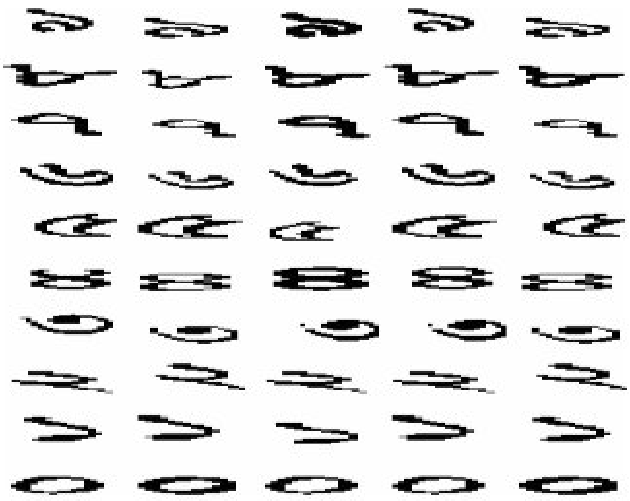

Fig. 4. Examples of training image data of Bengali numerals used to select structuring elements.

Fig. 5. Examples of scaled image data of Bengali numerals used to select structuring elements.

have the same alignment. Each of these rotated objects may be scaled by such a factor so that their upright minimal bounding boxes have the same dimensions. These geometric transformations of the training images should be preceded by noise cleaning. And this noise cleaning is done by opening and closing the input image by a small isotropic (disk) structuring element. After having these transformations we proceed to find the representative of each class. Some examples of training images used for generating the class representatives are shown in Fig. 4. Note that the training images are reasonably noise free. However, this may not be true always. Since these numerals are extracted from skew corrected document image, their orientations are also more or less normalized. However, we had to scale them to bring them to a fixed size (in our experiment it is 64×64). The scaled images are shown in Fig. 5. Then class representatives are determined by union and intersection among objects of each class. The results of these two operations are shown in Fig. 6. Finally, the extracted structuring elements are shown in Fig. 7.

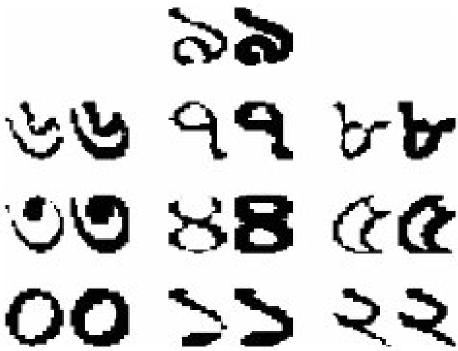

Fig. 6. Examples of class representatives of Bengali numeral classes used to select structuring elements.

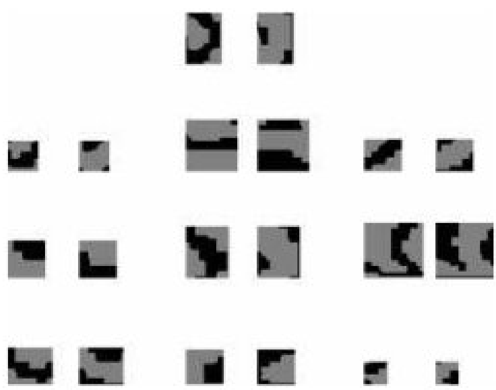

Fig. 7. Selected structuring element pairs (arranged in same order for objects in the previous figure, but not in the same scale) to be used to recognize Bengali numerals using hit-and-miss transformation.

When these selected structuring elements are used in the object recognition system using hit-and-miss transform, the test object too has to undergo similar geometric transformation. For the test images too, we need to clean noise in the similar way as in the case of training images.

5. Conclusion

In this paper we have proposed a novel technique for automatic selection of structuring element for a class of problems, namely object recognition. The method is used to develop a Bengali numeral recognition system using morphological hit-and-miss operation. To devise the algorithm a new measure of local shape is suggested. Based on this property minimal (in terms of area of domain of definition of the structuring element) distinctive structuring elements are extracted from the classes of objects in a deterministic way. The algorithm is computationally efficient and is guaranteed to terminate with the required output if the classes are distinct.

References

1. B. Chanda, "Application of binary morphology to separate overlapped objects," *Patt. Recogn. Lett.* **13** (1992) 639–645.
2. B. Chanda and R. Bhattacharya, "Automatic selection of structuring element for object classification through morphology," *Imag. Process. Commun.* **5**, 1 (1999) 67–79.
3. B. Chanda and D. Dutta Majumder, *Digital Image Processing and Analysis*, Prentice Hall of India, New Delhi, 2000.
4. C. R. Giardina and E. R. Daugherty, *Morphological Methods in Image and Signal Processing*, Prentice Hall, Englewood Cliffs, 1987.
5. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Vol. 1, Addison-Wesley, Reading, MA, 1992.
6. P. Jackway, W. Boles and M. Deriche, "Morphological scale-space fingerprints and their use in object recognition in range images," *Proc. 1994 IEEE Int. Conf. Acoustics Speech Signal Processing*, Adelaide, Australia, April 1994, pp. V5–8.
7. B. K. Jang and R. T. Chin, "Analysis of thinning algorithm using mathematical morphology," *IEEE Trans. Patt. Anal. Mach. Intell.* **PAMI-12** (1990) 541–557.
8. S. Loncaric and A. P. Dhawan, "Near-optimal MST-based shape description using genetic algorithm," *Patt. Recogn.* **28**, 4 (1995) 571–579.
9. J. R. Mandeville, "Novel method for analysis of printed circuit images," *IBM J. Res. Dev.* **29** (1985) 73–86.
10. J. B. T. M. Roerdink, "Computer vision and mathematical morphology," *Comput. Suppl.* **11** (1996) 131–148.
11. A. Rosenfeld and A. C. Kak, *Digital Picture Processing*, Vol. 2, Academic Press, NY, 1982.
12. J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, London, 1982.
13. L. G. Shapiro *et al.*, "Shape representation with mathematical morphology," *Proc. 8th IEEE Int. Conf. Pattern Recognition*, Paris, 1986, pp. 416–418.
14. D. Sinha and C. R. Giardina, "Discrete black and white object recognition via morphological functions," *IEEE Trans. Patt. Anal. Mach. Intell.* **PAMI-12** (1990) 275–293.

15. A. W. M. Smeulders and C. de Boer, "Design and performance in object recognition," *Lecture Notes in Computer Science: Graphics Recognition*, eds. K. Tombre and A. K. Chhabra, Springer, 1998, pp. 335–346.
16. S. R. Sternberg, "A morphological approach to finished surface inspection," *IEEE Trans. Acoust. Speech Sign. Process.* **ASSP-34** (1985) 1329–1332.

Bhabatosh Chanda received his B.E. in electronics and telecommunication engineering and Ph.D. in electrical engineering from the University of Calcutta in 1979 and 1988, respectively. He received the "Young Scientist Medal" of Indian National Science Academy in 1989, "Computer Engineering Division Medal" of the Institution of Engineers (India) in 1998 and "Vikram Sarabhai Research Award" of Physical Research Lab in 2002. He is also recepient of UN fellowship and Diamond Jubilee fellowship of National Academy of Sciences, India. He is a fellow of The Institution of Electronics and Telecommunication Engineers, and the National Academy of Sciences, India. He worked at Intelligent System Lab, University of Washington, Seattle, USA as a visiting faculty from 1995 to 1996. He has published more than 70 technical articles in refereed international journals.

His research interest includes image processing, pattern recognition, computer vision and mathematical morphology. He is currently working as a Professor in Indian Statistical Institute, Calcultta, India.

B. B. Chaudhuri received his B.Sc. (Hons), B.Tech and M.Tech degrees from Calcutta University, India in 1969, 1972 and 1974, respectively and Ph.D. degree from Indian Institute of Technology, Kanpur in 1980. He joined Indian Statistical Institute in 1978 where he served as the Project Coordinator and Head of National Nodal Center for Knowledge Based Computing funded by DoE and United Nations Development Program. Currently, he is the Head of Computer Vision and Pattern Recognition Unit of the Institute as well as head of Resource Centre for Indian Language Technology Solution, Bengali. He developed the first workable OCR for Bangla, Devnagari and Oriya, three major Indian scripts. Also, he developed a speech synthesis and a spell-checker systems in Bangla apart from doing many computational linguistic studies on Indian languages. He also developed software for Bio-medical Image Processing, that has been used in medical centres. He has published about 250 research papers in reputed international journals, conference proceedings and edited books. Also, he has authored three books entitled *Two Tone Image Processing and Recognition* (Wiley Eastern, 1993), *Object Oriented Programming: Fundamentals and Applications* (Prentice Hall, 1998) and *Computer and Software Technology Dictionary* (Ananda Publishers, 2002). He has been awarded Sir J. C. Bose Memorial Award for best engineering science oriented paper in 1986, M. N. Saha Memorial Award (twice) for best application oriented paper in 1989 and 1991, Homi Bhabha Fellowship award in 1992 for OCR of the Indian Languages and computer communication for the blind, Dr. Vikram Sarabhai Research Award in 1995 for his outstanding achievements in the fields of electronics, informatics and telematics, and C. Achuta Menon Prize in 1996 for computer-based Indian Language Processing. He has worked as a Leverhulme visiting fellow at Queen's University, U. K. in 1981–82, as a visiting scientist at GSF, Munich, and as a guest faculty at the Technical University of Hannover during 1986–88 and 1990–91. He is a member secretary (Indian Section) of International Academy of Sciences, Fellow of National Academy of Sciences (India), Fellow of Institution of Electronics and Telecommunication Engineering, and Fellow of the Indian National Academy of Engineering. He has served as a guest editor of a special issue of the *Journal of Institute of Electronics & Telecom Engineering* (*IETE*). Presently, he is serving as an associate editor of *Pattern Recognition* (Pergamon Press), *Pattern Recognition Letters* (Elsevier Sciences) and *VIVEK*. In 1999 he was elected as a Fellow of International Association of Pattern Recognition. In 2000 he has been honoured with the Fellowship of IEEE (Institute of Electrical and Electronics Engineers). He was chosen for this rare honour for his pioneering contributions to Pattern Recognition, especially Indian language script OCR, document processing and natural language processing.

His research interests include pattern recognition, image processing, computer vision, natural language processing and digital document processing.