Pergamon

PII:S0952-1976(97)00002-X

Engng Applic. Artif. Intell. Vol. 10, No. 2, pp. 213–224, 1997

0952-1976/97 $17.00+0.00

Contributed Paper

Using Curvature Features in a Multiclassifier OCR System

BASILIOS GATOS
Democritus University of Thrace, Xanthi, Greece
Institute of Informatics and Telecommunications, Athens, Greece

NIKOS PAPAMARKOS
Democritus University of Thrace, Xanthi, Greece

CHRISTODOULOS CHAMZAS
Democritus University of Thrace, Xanthi, Greece

(Received July 1996; in revised form December 1996)

This paper proposes a new method for character recognition which is based on a novel sequential multiclassifier scheme that uses curvature features in conjunction with other outstanding features. The curvature features are obtained according to the slopes of the character edge pixels. These slopes are observed from certain outer and inner character directions, are grouped according to slope type, and are evaluated according to the areas of the orthogonal triangles which have as hypotenuse the resulting slope lines. Through the multiclassifier technique, different feature types can be used for each character, and combined in a sequential way. Using this approach the recognition rate becomes greater in comparison with single classifiers. To improve the effectiveness of the multiclassifier, a technique has been developed that gives the optimum distance threshold value for each classifier in order to decide with certainty for a character. In this way, characters having very close feature vectors can be distinguished. The proposed method gives highly accurate results with low error rates even if a different recognition font set is used, or in the case of the recognition of deformed fonts. Experimental results with Latin and Greek characters show that the recognition rate can exceed 99.7%.

Keywords: Optical character recognition, feature extraction, multiple classifiers, thresholding.

1. INTRODUCTION

The processing of typewritten texts is of great interest in office automation. Optical Character Recognition (OCR) techniques are used to translate human readable characters into machine readable codes. An excellent survey on OCR is given by Impedovo *et al.* (1991). The choice of appropriate features in OCR techniques is of course of the utmost importance. Until now, several OCR techniques based on statistical, matching, transform and shape features have been proposed with varying success (Cash and Hatamian, 1987; Kahan *et al.*, 1987; Papamarkos *et al.*, 1994). On the other hand, it is well known that a set of "good" features generally embodies some important characteristics such as:

- *Discrimination.* Features should take on significantly different values for characters belonging to different classes.
- *Reliability.* Features should take on similar values for characters belonging to the same class.
- *Independence.* Features should be uncorrelated with each other.
- *Small feature space.* The number of features should be small to make classification simple and fast.

Additionally, the features must satisfy other desirable requirements such as low computational cost and low complexity of the feature-extraction techniques. For these reasons, simple and powerful features cannot be easily found.

Correspondence should be sent to: Assoc. Prof. Nikos Papamarkos, Electric Circuits Analysis Laboratory, Department of Electrical & Computer Engineering, Democritus University of Thrace, 67100 Xanthi, Greece [E-mail: papamark@voreas.ee.duth.gr].

Many integrated OCR systems have been proposed, combining a feature-extraction technique with a classifier scheme, such as: features based on the application of the distance transform in gray-scale image in combination with a k-Nearest-Neighbor classifier (Kovacs and Guerrieri, 1995), line and junction extraction as features and Gaussian statistical classifier (Fleming and Hemmings, 1983), endpoints, corners and tees extraction as features and a binary decision tree classifier (Kerrick and Bovik, 1988; Gatos and Papamarkos, 1993), shape clustering feature extraction and statistical Bayesian classifier (Kahan *et al.*, 1987), Fourier and topological descriptors as features and statistical distance measures as a classifier (Shridhar and Badreldin, 1984).

The proposed paper describes a new method for OCR which is based on the curvatures of the characters (Gatos and Papamarkos, 1995b). Specifically, after preprocessing and segmentation, the characters' curvature characteristics are obtained by using the slopes of the characters' edge pixels. These slopes are observed from the Left, Right, Up and Down directions, and from outer and inner positions of the characters. The result of the above procedure is a set of curvature features which are grouped according to slope type, and are evaluated according to the areas of the triangles which have as hypotenuse the resulting slope lines. Since it is possible to have from each direction several curvature characteristics of the same type, features of the characters are specified by all the similar characteristics prevailing in certain zones, in which the character has been separated horizontally and vertically. Following this procedure, a 48-element feature vector is extracted for each character. As is shown by the experimental results, curvature features give better results in single-font or omnifont experiments, compared with other well-known features such as Zernike moments and Gaussian masks.

Several authors have tried to combine the information extracted from different feature extraction and classifier techniques (Xu and Krzyzak, 1992; Rogova, 1994; Battiti and Colla, 1994). Most of these approaches use a parallel multiclassifier scheme. In order to preserve the high speed of the finally constructed system, a sequential multiclassifier OCR scheme is proposed that calls to a next classifier only if it cannot get a recognition result with certainty from the previous classifiers. To design this system, a condition is defined which determines whether a classifier decides with certainty for a character. In order to overcome cases with characters having very close feature vectors, a technique was developed for optimizing the previously defined condition. Considering the discrimination information of different patterns of the training set, a classifier can decide on a character only if there is a training pattern close enough, with great discrimination ability compared with other patterns of the training set. For the same multiclassifier system but without on-line learning, a new procedure is applied for selecting the best result from all the classifiers, when there is no result with great certainty from any of them. In order to design and test the multiclassifier system, a character-validation set is used, that gives statistical information about the efficiency of every classifier. The system was tested with several font types, including Greek and Latin characters, and the experimental results show that in many cases the recognition rate can be greater than 99.7%.

2. PREPROCESSING

Starting with a digitized document, a preprocessing scheme is necessary. This preprocessing operation includes the following main procedures:

- An optimal threshold technique to obtain a binary image for the digitized document. For this stage, the new threshold technique is used, which is based on an algorithm proposed by Papamarkos and Gatos (1994). This technique is simple and gives satisfactory binary images.
- Rotation of the document to remove any skewing variation by using the information that exists on multiple vertical lines of the text image (Gatos and Papamarkos, 1995a; Gatos *et al.*, to appear). Due to the text skew, each horizontal text line intersects the vertical lines at non-horizontal positions. Using these intersections a correlation matrix is constructed, whose global histogram maximum provides the skew angle of the document. After this step, all symbols have approximately the same orientation.
- Image filtering to reduce the noise in the image, and to achieve optimal boundary extraction. As a final filtering process, morphological filters are applied in order to suppress small islands and sharp caps of the characters (Jain, 1989; Gonzalez and Woods, 1993).
- Segmentation techniques to separate the characters from the background (Fujisawa *et al.*, 1992).

For the proposed OCR system a top-down unsupervised segmentation method has been developed, which consists of three main stages. In the first stage, the method determines the document's major blocks, in the second stage it proceeds to the text-line extraction, and finally the method terminates with the character-separation stage. For the first and second stages, where the Run-Length Segmentation Algorithm (RLSA) (Wong *et al.*, 1982; Wahl *et al.*, 1982; Wang and Shihari, 1989) is used, an algorithm has been developed for the automated calculation of the proper horizontal and vertical smoothing values. Specifically, the horizontal and vertical smoothing values are calculated according to the horizontal and vertical distributions of the black and white run-lengths. The values of the mean character length and the mean text line distance are determined using the contributions of the horizontal and vertical run-lengths.

For the final stage, a new fast algorithm for character segmentation is applied. This algorithm has been developed to overcome the segmentation problems associated with overlapping characters. This is a fast algorithm, that ensures, with 100% performance, proper separation of normal type or overlapping characters. The general idea is

Fig. 1. Priorities: (a) upper direction, (b) lower direction.

to detect appropriate boundaries, consisting of vertical line segments, on both the right- and left-hand sides of each character.

These boundaries are known as "routes". For each character, the method searches to find the right-hand route, since the left-hand one is already known from the previous process. This is done by moving from the known left-hand route to the right, until a black pixel is found. This pixel is the starting point for the separation process of the next character, and is designated A. Next, the search is continued horizontally through the character's pixels until a white pixel is found (this pixel node is designated B). From this pixel, the character boundary is followed, first in the upper and then in the lower direction. The aim of this procedure is the discovery of two sub-routes that can construct the right-hand route of the character. In the upper direction, movement takes place pixel by pixel, with the priority a-b-c [Fig. 1(a)] until either the top of the character is reached, or an obstacle that stops the movement [see Fig. 2(a)]. If this happens, the path moves horizontally to the right, through the character's pixels, and searches to find a new white pixel, i.e. a new node. Then, the vertical movement is resumed, and continued until the top is reached. Note that during the upper direction movement no previously used pixel is revisited, that is, the path passes only once through each pixel belonging to it.

When the upper sub-route has been found, the algorithm returns to the last node pixel, and starts to search for the lower sub-route. The movements follow the priorities of Fig. 1(b), and stop if the bottom of the character is reached.

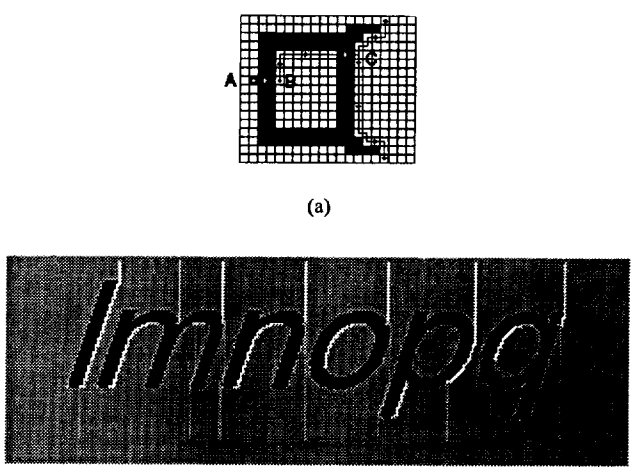

Fig. 2. Application of the character-separation procedure for the character α and for a word.

As in the vertical movement, in the case of an obstruction, the path moves horizontally to the right to find a new node. When both the upper and lower sub-routes have been detected, the character is separated out. The character-separation procedure is continued until the end of the text line is reached.

The advantages of this method, besides its speed, are that it can keep any accent-marks together with the separated characters. Thus, letters such as i and j will not lose their identity, as they would if the accent-mark were removed by the segmentation process. Furthermore, characters consisting of more than one piece, like the Greek letter Ξ, will remain unbroken after segmentation. Figure 2 shows the application of the character-separation procedure for the character α, and for a word.

3. FEATURE EXTRACTION

For the extraction of the curvature features, it is necessary to determine the characteristic slopes, observed from specific horizontal and vertical directions. Each character is normalized and fitted in a $M \times N$ array I_m. To evaluate the character slopes, first the sequences Q_n, $n=1, 2, \ldots, 8$ are determined, based on character edge points. These sequences are formed using the distances of the character edge points from the left or "up" array edges. For every edge point, a slope is evaluated, all the slopes of the same type are grouped together, and characteristic orthogonal triangles are formed. To construct these triangles, their hypotenuses are first determined by grouping neighbouring slopes of the same type. The curvature features are then extracted by calculating the areas of the characteristic triangles prevailing in certain zones in which the character is separated horizontally and vertically. The areas of these triangles depend on the lengths as well as on the slopes of their hypotenuses, so it can be stated that these features represent the characters' curvature characteristics.

The feature-extraction procedure is explained in detail in this section.

3.1. Calculation of the characteristic sequences

The first stage of the curvature feature-extraction procedure is the determination of the characteristic slopes, observed from specific horizontal and vertical directions. The observed directions for those slopes depend on the position of the observer. As can be seen in Fig. 3(a) eight observer directions are specified. The edge points an observer can see in any of those directions are defined as characteristic points. Sequences Q_n, $n=1, \ldots, 8$ contain the distances of the characteristic points from the left or "up" character array edges, and for horizontal or vertical observation, respectively. Table 1 gives the relation of variable n of Q_n with the eight possible observer positions.

Figure 4 shows a character stored in the matrix I_m, and its eight sequences Q_n. For example, sequence Q_1 corresponds to the distances from the left-hand array edge of all the points which the observer LO can see, from top to bottom [Fig. 4(c)]. At line 0 the observer cannot see a pixel, so the

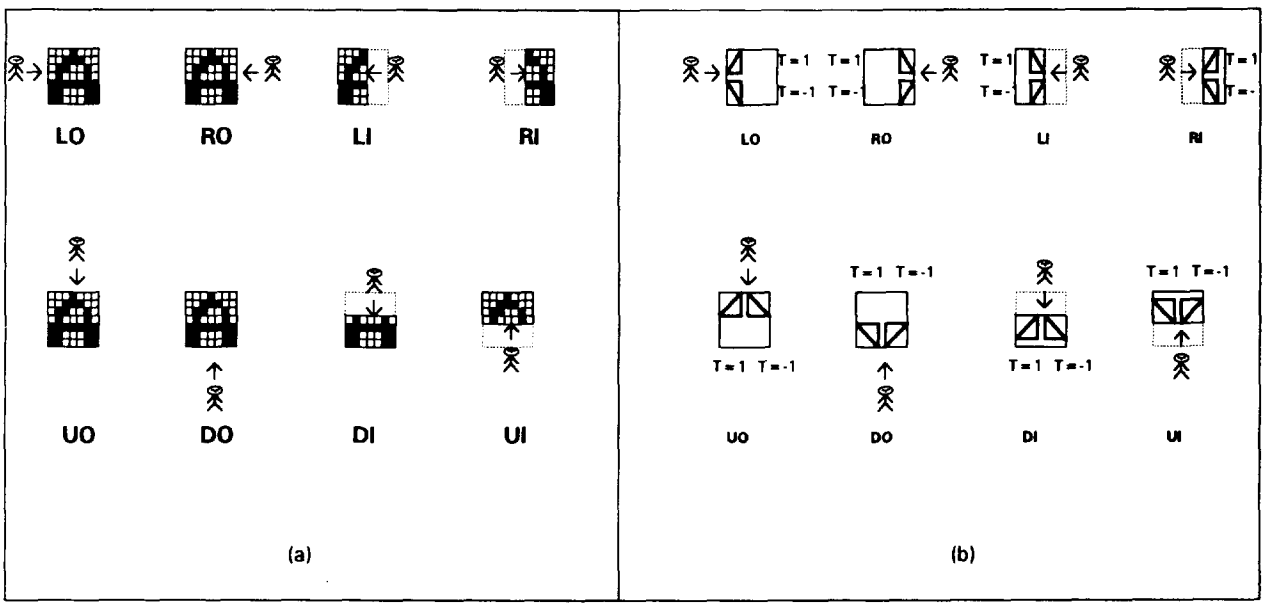

Fig. 3. The eight observer directions: Left Out (LO), Right Out (RO), Up Out (UO), Down Out (DO), Left In (LI), Right In (RI), Up In (UI) and Down In (DI). For every observer position there are two triangle types (b) for $T=1,-1$.

maximum value, 8, is stored. Next, the observer can see the pixel having distance 4 from the left-hand edge, and then the pixel having distance 3.

3.2. Calculation of the characteristic slopes

For every characteristic point a slope is calculated, which depends on the correlation of the position of that point with the position of the next and previous points in the sequence Q_n. From the eight sequences Q_n and for every n, the eight slope sequences S_n are derived as follows:

$$S_n=\begin{cases} \text{I or } - & \text{unchanging slope} \\ / & \text{decreasing slope} \\ \backslash & \text{increasing slope} \\ \# & \text{slope gap} \end{cases} \quad (1)$$

According to their previous and next points, each characteristic point is assigned to a slope according to Table 2. Figure 4(c) shows how the sequence S_1 is derived from Q_1. For example, $S_1(2)$="/" because the sequence Q_1 decreases from the previous point $Q_1(1)$ to the next point $Q_1(3)$, that is, $Q_1(1)>Q_1(2)>Q_1(3)$.

Table 1. The relation of the variable n of the sequence Q_n with the eight observation position directions

Observer position direction	n
LO (Left Out)	1
RO (Right Out)	2
UO (Up Out)	3
DO (Down Out)	4
LI (Left In)	5
RI (Right In)	6
UI (Up In)	7
DI (Down In)	8

3.3. Calculation of the characteristic triangles

Using the sequence of the slopes characteristic triangles can be calculated by grouping successive same slopes. For every sequence S_n, the procedure starts from a certain slope $S_n(a)$ and searches successively until an opposite slope is detected, or a slope gap corresponding to $S_n(b)$. From this procedure a characteristic orthogonal triangle is obtained, having as hypotenuse the line segment that is defined from pixels that correspond to $Q_n(a)$ and $Q_n(b-1)$. A simple example of this procedure is given in Fig. 4(c). The procedure starts from $S_1(1)$ with a decreasing slope, and then examines the sequence S_1 successively until the increasing slope of $S_1(7)$ is found. Doing this gives the characteristic triangle that is derived from the pixels that correspond to $Q_1(1)$ and $Q_1(6)$. The area of these characteristic triangles is $0.5|Q_n(b-1)-Q_n(a)|(b-a-1)$. According to the slope type, there are two triangle types for every observer position [Fig. 3(b)].

3.4. Extraction of curvature features

As was mentioned earlier, the areas of these orthogonal triangles correspond to the length and the slope of their hypotenuses, and can be taken as curvature features. To classify characters according to the areas of their characteristic triangles, the areas of the characteristic triangles of the same type prevailing in certain zones, in which the character has been separated horizontally and vertically, are added. Each character is divided in three equal-distance horizontal and three equal-distance vertical zones. In each zone, the areas of the orthogonal triangles of the same type are added. For every triangle type and for every observer direction

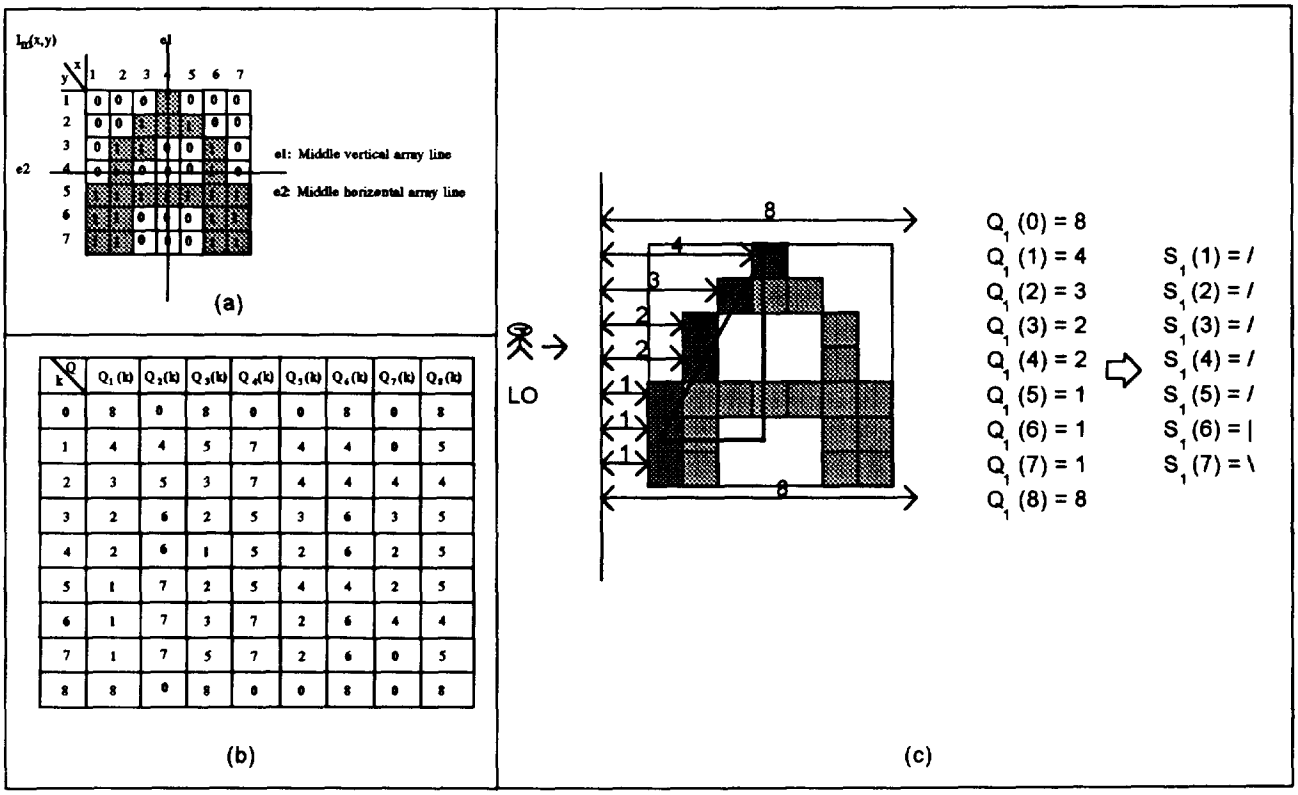

Fig. 4. The normalized Im character array (a) and the Q sequences of its characteristic points (b). Figure (c) depicts the calculation of the sequences Q_1 and S_1 of the character.

three features are formulated according to the existing zone, of the following type:

$$f_{T,O,1} = \sum_{i=1\ldots i_{\max}} \frac{1}{2} |Q_n(b_i - 1) - Q_n(a_i)|(b_i - a_i - 1)$$

$$\text{if } a_i, b_i \leq p/3 \tag{2}$$

$$f_{T,O,2} = \sum_{i=1\ldots i_{\max}} \frac{1}{2} |Q_n(b_i - 1) - Q_n(a_i)|(b_i - a_i - 1)$$

$$\text{if } a_i, b_i > p/3 \wedge a_i, b_i < 2p/3$$

$$f_{T,O,3} = \sum_{i=1\ldots i_{\max}} \frac{1}{2} |Q_n(b_i - 1) - Q_n(a_i)|(b_i - a_i - 1)$$

$$\text{if } a_i, b_i \geq 2p/3$$

where $p = x_{\max}$ for Q_1, Q_2, Q_5, Q_6 and $p = y_{\max}$ for Q_3, Q_4, Q_7, Q_8; $T = -1$ or 1 due to triangle type [Fig. 1(b)]; $O = 1, 2, \ldots, 8$ due to observer position [Fig. 1(a)]; $i_{\max}$ = the number of triangles of the same type.

The total number of curvature features is $2 \cdot 8 \cdot 3 = 48$ (2 triangle types, 8 observer directions, 3 character zones). Figure 5 shows the 48 types of feature for the character "δ".

4. USING A SEQUENTIAL MULTICLASSIFIER

When a character matrix is transformed to a feature vector, only the essential information from the character is selected, that will help with its classification. Because this information varies among feature-extraction schemes, it can be stated that a combination of several classifiers with different feature-extraction schemes can lead to high recognition rates, even in cases where each classifier alone may fail. In order to preserve high-speed operation, a sequential rather than a parallel multiclassifier type was selected here. The reason is that, in many cases, a classifier can decide with certainty, so there is no need to spend time on employing other classifiers with other feature-extraction schemes.

The main scheme of a sequential multiclassifier system is demonstrated in Fig. 6(a). Three independent classifiers are used in this application. For each classifier, a different feature-extraction scheme is used. Initially, each feature vector is directed to the first classifier. If the classifier cannot decide with certainty, the next classifier is called; if none of the classifiers can decide with certainty, the algorithm proceeds to the learning phase. In the same way, this scheme can be extended for more than three classifiers. When a non-iterative OCR system is used without a learning phase, the construction of the two-stage classifier shown in Fig. 6(b) is proposed. The first stage consists of the sequential multiclassifier technique, and the second stage (which will be used only in cases of uncertain results from the first stage) is the decision as to the best result, using the results of all the classifiers. Clearly, the sequential multiclassifier algorithm with a learning stage has better recognition rates compared to the non-iterative one. The

multiclassifier system uses two decision rules. Each character is recognized as a decision of a specific classifier, or else as the best solution derived by the entire sequential classifier.

4.1. Design of the sequential multiclassifier system

A classifier is trained using a learning set of FMAX patterns, each one with PMAX features. Every pattern i of the learning set has the feature vector $F_i=[f_{1,i}, f_{2,i}, \ldots, f_{PMAX,i}]$, $i=1, \ldots,$ FMAX, and belongs to the class FP_i. It is necessary to decide whether a character with feature vector $T=[t_1, t_2, \ldots, t_{PMAX}]$ can be classified with certainty, or if it must be forwarded to the next classifier. In order to get statistical information about the efficiency of the classifier, a validation set of VMAX patterns is used. Every pattern j of the validation set has a feature vector $V_j=[v_{1,j}, v_{2,j}, \ldots, v_{PMAX,j}]$, $j=1, \ldots,$ VMAX, and belongs to class VP_j. The distance of a validation pattern V_j from a learning pattern F_i is:

$$D(V_j,F_i)=\sum_{k=1}^{PMAX} |v_{k,j}-f_{k,i}|. \quad (3)$$

A validation pattern is classified to the class $R(V_j)$ according to the formula:

$$R(V_j)=p: D(V_j,F_i)=\text{minimum for } i=m \text{ where } FP_m=p. \quad (4)$$

$COR(V_j)$ is defined as the function of correct classification as follows:

$$COR(V_j)=\begin{cases}1, & \text{if } R(V_j)=VP_j\\ 0, & \text{otherwise.}\end{cases} \quad (5)$$

The classifier can decide on the classification of the validation pattern V_j only if there is a learning pattern that is fairly close to it. Using a distance limit Th, it is defined that the classifier can make a decision for pattern V_j only if the function $THR(V_j,Th)$ takes value 1:

$$THR(V_j,Th)=\begin{cases}1, & \text{if } \underset{i=1..FMAX}{\text{minimum}} (D(V_j,F_i))<Th\\ 0, & \text{otherwise.}\end{cases} \quad (6)$$

According to the threshold Th, the classifier can make a decision for a part of the validation set which is given by the Processed Characters function $PC(Th)$:

Table 2. The values of the characteristic points' slope sequences S according to characteristic point sequences Q

Q_1 Q_2 Q_5 Q_6	S_1 S_2 S_5 S_6	Q_3 Q_4 Q_7 Q_8	S_3 S_4 S_7 S_8	Q_3 Q_4 Q_7 Q_8	S_3 S_4 S_7 S_8
k-1 k k+1	|	k-1 k k+1	—	k-1 k k+1	/
k-1 k k+1	|	k-1 k k+1	—	k-1 k k+1	/
k-1 k k+1	|	k-1 k k+1	—	k-1 k k+1	\
k-1 k k+1	/	k-1 k k+1	—	k-1 k k+1	\
k-1 k k+1	/	k-1 k k+1	—	k-1 k k+1	\
k-1 k k+1	/	k-1 k k+1	—	k-1 k k+1	\
k-1 k k+1	\	k-1 k k+1	/	k-1 k k+1	\
k-1 k k+1	\	k-1 k k+1	/	k-1 k k+1	#
k-1 k k+1	#	k-1 k k+1		k-1 k k+1	

$$PC(Th)=100\frac{\sum_{j=1}^{VMAX}THR(V_j,Th)}{VMAX}\%. \quad (7)$$

The percentage of correct classifications (recognition rate) of the part of the validation set that is processed by the classifier is given by the function $RR(Th)$ as follows:

$$RR(Th)=100\frac{\sum_{j=1}^{VMAX}(THR(V_j,Th)COR(V_j)}{\sum_{j=1}^{VMAX}THR(V_j,Th)}\%. \quad (8)$$

Th_{opt} is defined as the optimum distance limit for deciding whether a pattern from the validation set will be classified by the specific classifier, or if it will be forwarded to the next classifier. The value of Th_{opt} is evaluated from the condition according to which 100% recognition is obtained for the maximum number of characters of the validation set:

$$Th_{opt}=Th: RR(Th)=100\%\wedge PC(Th)=\text{maximum}. \quad (9)$$

Now, a character with feature vector T is well classified only if $\text{minimum}_{i=1\ldots FMAX}(D(T,F_i))<Th_{opt}$.

Figure 7(a) shows the variety of functions $PC(Th)$ and $RR(Th)$ for a certain validation set. For this example, according to equation (9), the optimum value of Th is 140. It can be observed that this corresponds to 19.04% of the validation set patterns.

4.2. Optimizing the multiclassifier using discrimination information

According to the above-mentioned technique, a classifier can decide only for those characters that lie at a short distance from a learning pattern. Unfortunately, there are some cases where a classifier confuses similar characters, especially when dealing with Greek characters. For example, say a classifier has been trained for the similar characters "ν" and "υ". A new character that belongs to class "ν" is forwarded to the classifier. However small the distance of Th_{opt}, possible wrong classifications to class "υ" cannot be prevented, for the reason that the character will be misclassified to class "υ" with even a short distance. This happens because during the training phase it is difficult to take into account that characters "ν" and "υ" can have a very short distance between their feature vectors.

In order to handle cases of characters with very close feature vectors, all the above formulae for the sequential multiclassifier OCR system must be readjusted. For every pattern i of the learning set, a discrimination ability function $DIS(i)$ is defined, that gives its minimum distance from all the patterns in the learning set that belong to different classes of pattern i:

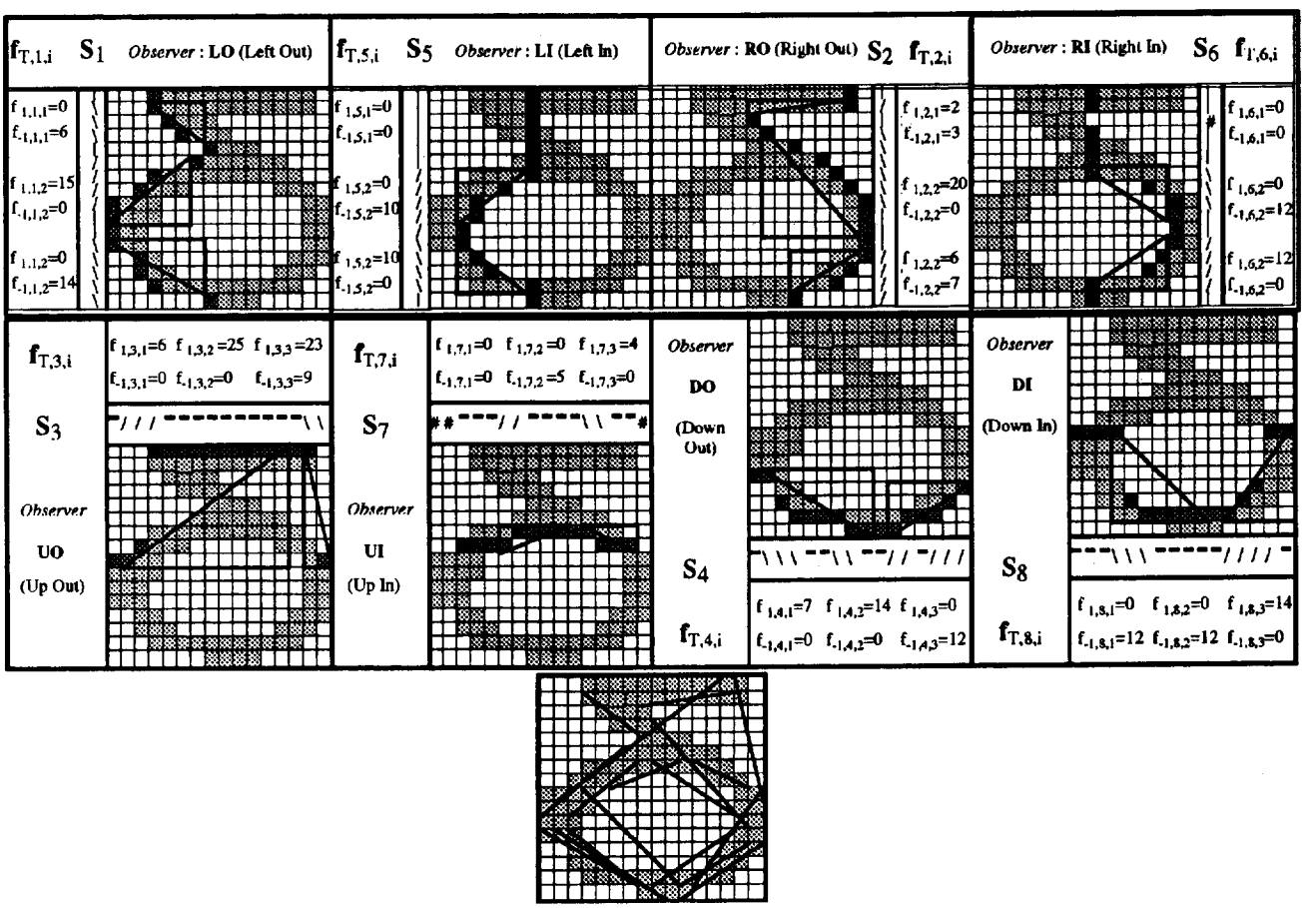

Fig. 5. The 48 curvature features of the letter δ and the correspondence of the character with the hypoteneuses of its characteristic triangles.

$$\mathrm{DIS}(i)=\underset{\substack{j=1,2,\ldots,\mathrm{FMAX}\\ j:\,\mathrm{FP}_j\neq\mathrm{FP}_i}}{\text{minimum}}\,(D(F_i,F_j)),\ i=1,2,\ldots,\mathrm{FMAX}. \quad (10)$$

As the value of the discrimination ability function DIS(i) increases, the discrimination of pattern i is better, compared with the other patterns. The normalized discrimination ability function DISN(i) $\in[0,1]$ is provided in the formula:

$$\mathrm{DISN}(i)=\frac{\mathrm{DIS}(i)}{\underset{j=1,2,\ldots,\mathrm{FMAX}}{\text{maximum}}\,(\mathrm{DIS}(j))},\ i=1,2,\ldots,\mathrm{FMAX}. \quad (11)$$

It is necessary to maximize the distance of a character from a learning pattern when the normalized discrimination ability function of the learning pattern has a small value. To achieve that, every distance function D (equation (3)) must be divided by the normalized discrimination ability function DISN(i). So, the formula, equation (6) that leads to the decision as to whether or not a classifier can classify a character, becomes:

$$\mathrm{THR}(V_j,Th)=\begin{cases}1, & \text{if } \underset{i=1,\ldots,\mathrm{FMAX}}{\text{minimum}}\left(\dfrac{D(V_j,F_i)}{\mathrm{DISN}(i)}\right)<Th\\ 0, & \text{otherwise.}\end{cases} \quad (12)$$

The condition that enables a classification result to be obtained from a certain classifier becomes:

$$\underset{i=1,\ldots,\mathrm{FMAX}}{\text{minimum}}\left(\frac{D(T,F_i)}{\mathrm{DISN}(i)}\right)<Th_{\mathrm{opt}} \quad (13)$$

where the new Th_{opt} is calculated again from equation (9).

Figure 7(b) shows the functions PC(Th) and RR(Th) of a validation set using the discrimination ability information of the learning set. According to equation (9) the optimum distance limit Th_{opt} is 310. It can be observed that using this limit, 33.33% of the validation set patterns are processed,

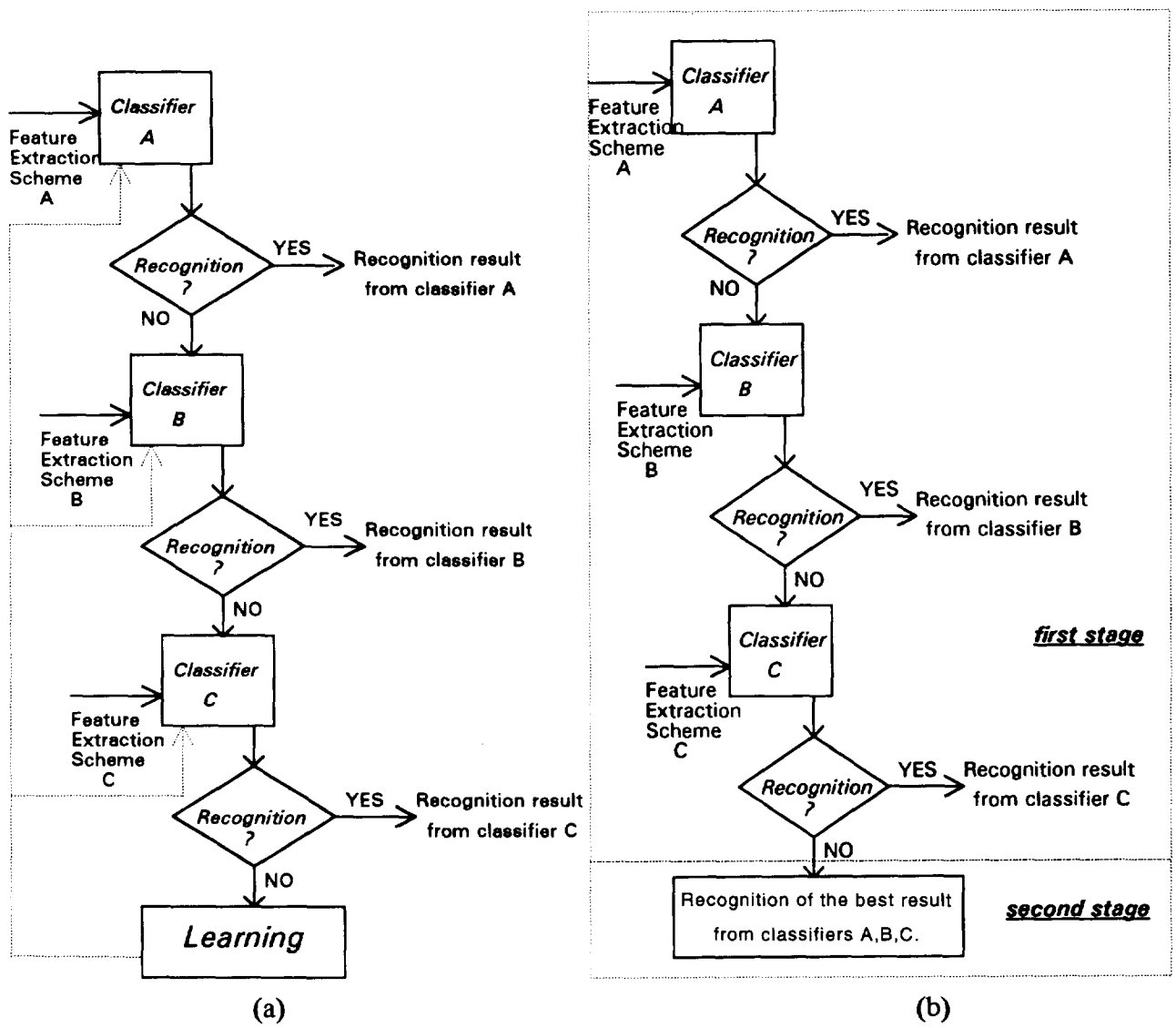

Fig. 6. Multiclassifier system consisting of three classifiers, with (a) and without (b) on-line learning.

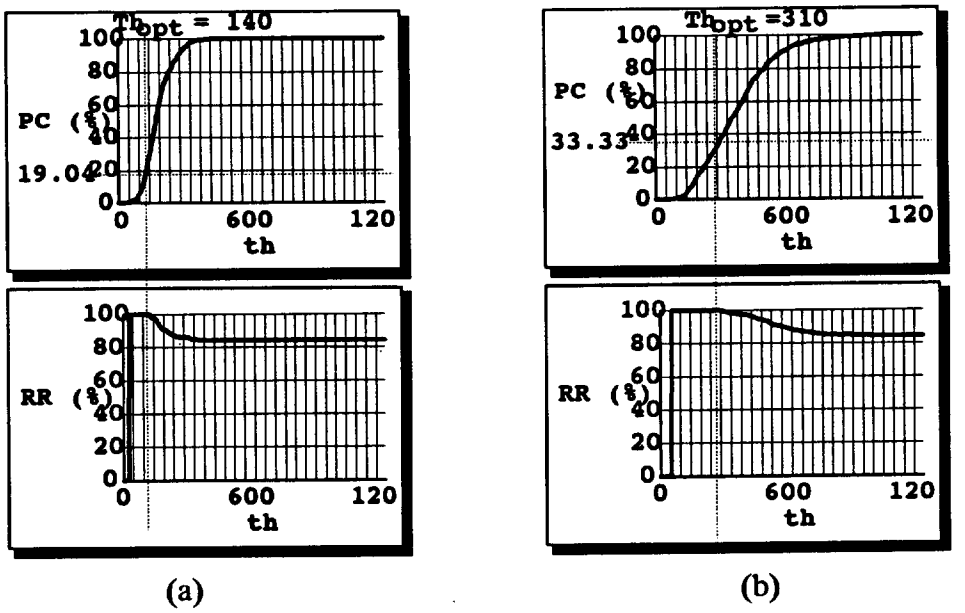

Fig. 7. Validation results for estimation of the optimum distance limit Th_{opt}, using the discrimination ability function (b) or not using it (a).

instead of the 19.04% that were processed without using the discrimination ability information of the learning set. Therefore, the use of the discrimination ability information of the learning set contributes to maximizing the credibility, the effectiveness and the speed of the sequential multiclassifier OCR system.

4.3. Designing a system without on-line learning

If the system is designed without a learning stage, it is essential always to have convergence of a character to a specific pattern of one of the classifiers that construct the multiclassifier system. To achieve this, a two-stage system is used [Fig. 6(b)]. The first stage is the sequential multiclassifier mentioned above, using a distance limit Th_{opt} for every classifier. If none of the classifiers can give a result with certainty, then the next stage is the choice of the best of all the classifier results.

In order to find the best classification results, for every classifier an efficiency function EFF is defined, which depends on how close the classifier distance limit Th_{opt} is to the minimum distance of the character from the classifier learning patterns:

$$\mathrm{EFF}=\frac{Th_{opt}}{\underset{i=1,\ldots,P_{max}}{\text{minimum}}\left(\frac{D(T,F_i)}{\mathrm{DISN}(i)}\right)-Th_{opt}}. \qquad (14)$$

So, in case of a multiclassifier system without on-line learning, the result of the classifier corresponds to the maximum value of the efficiency function EFF.

5. EXPERIMENTAL RESULTS

The proposed OCR method has been tested with various character sets in order to prove the efficiency of this novel feature-extraction method, as well as the efficiency of the multiclassifier technique. The characters' Zernike moments (Khotanzad and Hong, 1990) and the results of the application of overlapping Gaussian masks at the character surface are also used as features (Gatos *et al.*, 1993). The experimental procedure used a 300 dpi scanner resolution, 25×25 character normalization, 49 Zernike moment features (Zernike moments order=12) and 41 Gaussian mask features (25 5×5 and 16 4×4 masks were applied).

5.1. Experiment 1

In this experiment, the curvature features are compared with other outstanding feature-extraction schemes. In order to show the effectiveness of the multiclassifier OCR system, the training set used was a small set of 1152 Greek characters (32 classes) of Arc, Arial and Courier fonts, provided by MS-Windows True Type fonts and printed by an HP laser printer. The testing set used was:

- a set (4608 characters) of the same document fonts (Arc, Arial, Courier), printed by a laser printer;
- a set of the same fonts, printed by a pinovia 24-pin printer (3840 characters);
- a set of laser-printed bold characters of the same fonts (1920 characters); and
- a set of laser-printed characters of a different font (3840 characters from a Chi-Writer font).

These learning and testing set combinations were used with Zernike moments, Gaussian masks and a Curvature features single classifier scheme, and the results are demonstrated in Fig. 8(a), (b), (c) and (d), respectively. As is shown by these figures, Zernike moments perform well only if the same test font is used, and a laser printer output

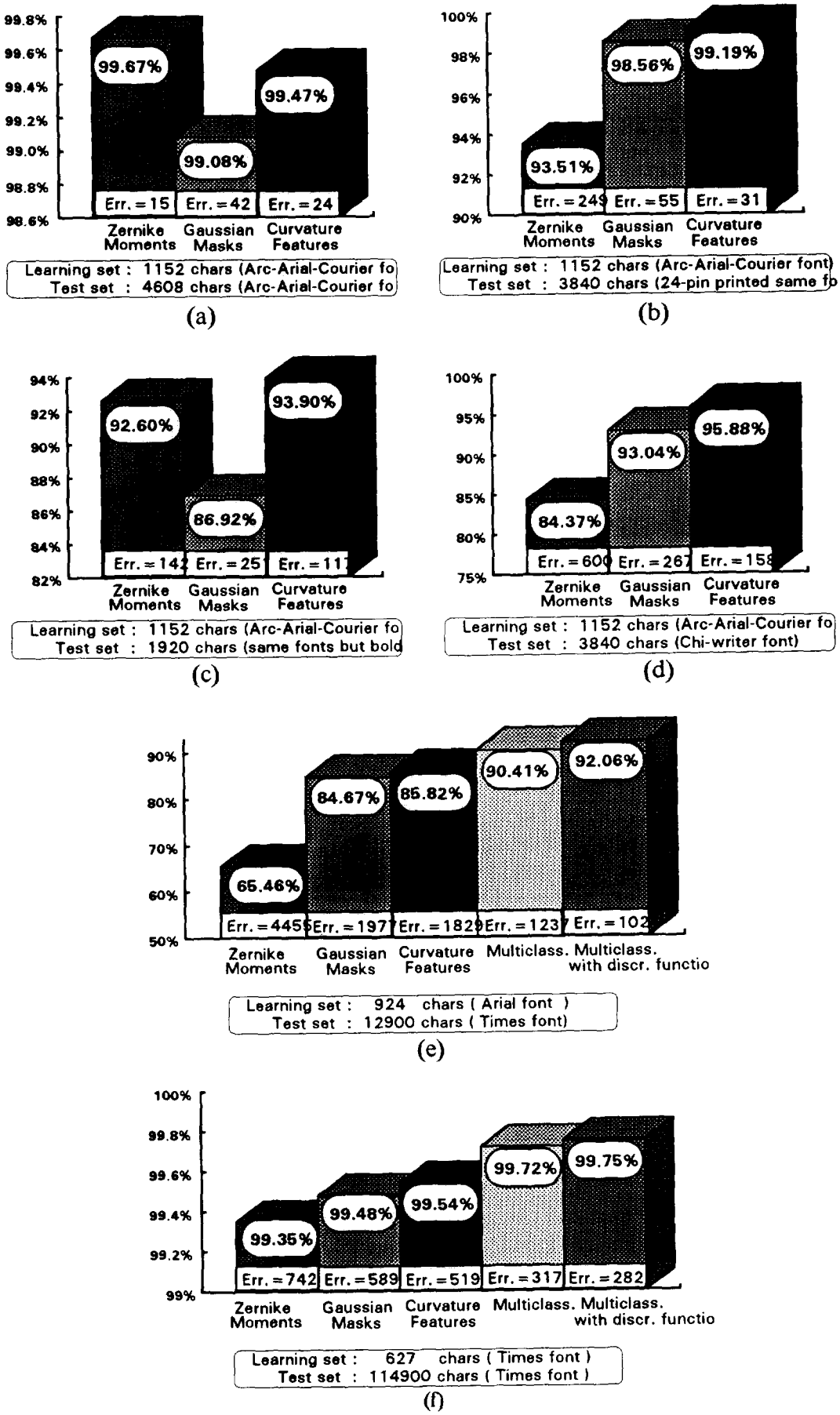

Fig. 8. Experimental results.

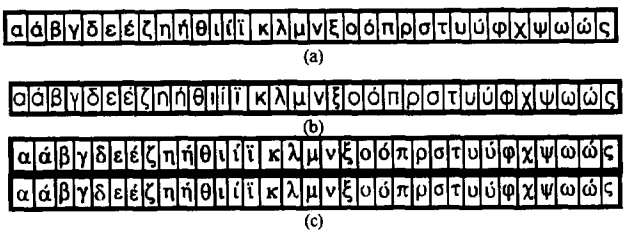

Fig. 9. Samples from the learning (a), validation (b) and test (c) sets.

device; Curvature features maintain high results when the font changes and a 24-pin printer is used, and give the best results in almost all the cases; Gaussian masks, although they give good results for font changes or a 24-pin printer, always give lower results compared with the curvature features.

5.2. Experiment 2

This experiment tests the efficiency of the multiclassifying system in a large and low-quality character set. Therefore, different fonts were applied for learning, validating and testing. The training set used was a total of 924 Greek characters (33 classes) of an Arial laser-printed font. As the test set, a large set of 12900 characters of Times laser-printed fonts was used. The results for Zernike moments, Gaussian masks and the curvature features single classifier schemes were compared, as well as the results of a multiclassifier scheme using all three feature-extraction schemes, with and without the discrimination ability function. For the multiclassifier construction a validation set of 924 Greek characters of a different font (Arc font) was used. The learning, validation and test fonts of this experiment are shown in Fig. 9. The results are demonstrated in Fig. 8(e), and show that: Zernike moments give low results because of font changes; the curvature features single classifier gives the best results compared with the other single classifiers; the multiclassifier technique improves on the single-classifier recognition results, the use of a discrimination function improves the efficiency of the multiclassifier, and gives the best results.

5.3. Experiment 3

In this experiment, the efficiency of a multiclassifier scheme was tested in cases with the same fonts in both at learning and testing stages, and for a very low-quality test set. As the training set, a total of 627 Greek characters (33 classes) of Times laser-printed fonts is used. As the test set, a large set of 114900 characters of the same font [Fig. 9(c)] was applied. The results with Zernike moments, Gaussian masks and the curvature features single-classifier schemes were compared, as well as the results of a multiclassifier scheme using the three feature-extraction schemes, with and without the discrimination ability function. The validation set consists of 627 Greek characters of the same font (Times font). The results are demonstrated in Fig. 8(f) and show that: curvature features give the best single-classifier results; and the multiclassifier technique using a discrimination function gives the best result, with a reliable recognition rate of 99.75%.

6. CONCLUSIONS

This paper proposes an OCR system which is based on a novel sequential multiclassifier scheme. The construction of the multiclassifier takes advantage of discrimination information of the separable feature sets, and can therefore classify efficiently even characters with very close feature vectors.

In order to achieve a high recognition rate, a new set of curvature features is proposed as a first feature set. This set corresponds to the areas of characteristic orthogonal triangles fitted in the characters' bodies, and effectively describes the curvatures of the characters' shapes.

The experimental results show that:

- In almost all single-classifier experiments, curvature features give the best results, compared with Zernike moments and Gaussian masks.
- Zernike moments perform well only in cases with no font changes or deformed fonts.
- The recognition rate using a single classifier system with curvature features is high, even with a different font style or characters from a low-quality printing device.
- The proposed sequential multiclassifier scheme increases the final recognition rate drastically, in both the omnifont and single-font experiments.
- The use of the discrimination ability function significantly improves the operation of the multiclassifier technique.

REFERENCES

Battiti, R. and Colla, A. M. (1994) Democracy in neural nets: voting schemes for classification. *Neural Networks*, **7**(4), 691–707.

Cash, G. L. and Hatamian, M. (1987) Optical character recognition by the method of moments. *Computer Vision, Graphics, and Image Processing*, **39**. 291–310.

Fleming, J. F. and Hemmings, R. F. (1983) A method of recognition for handwritten block capitals. *Pattern Recognition Letters*, **1**, 457–464.

Fujisawa, H., Nakano, Y. and Kurino, K. (1992) Segmentation methods for character recognition: From segmentation to document structure analysis. *Proc. of IEEE*, July, pp. 1079–1092.

Gatos, B. and Papamarkos, N. (1993) A Novel Method for Character Recognition. *Proc. of the 4th International Conference on Advances in Communication & Control (COMCON 4)*, pp. 493–503.

Gatos, B. and Papamarkos, N. (1995a) Skew detection in digitized documents. *Proc. of the 5th International Conference on Advances in Communication & Control (COMCON 5)*, Rethymno, Greece, pp. 124–130.

Gatos, B. and Papamarkos, N. (1995b) On the Development of an Ocr System Based on Curvature Features. *Proc. of the IEEE International Symposium on Industrial Electronics (ISIE'95)*, Athens, Greece, pp. 205–210.

Gatos, B., Karras, D. and Perantonis, S. (1993) Optical Character Recognition using Novel Feature Extraction and Neural Networks Classification Techniques. *Proc. of the Workshop on Neural Networks: Techniques and Applications*, Liverpool.

Gatos, B., Papamarkos, N. and Chamzas, C. Skew detection and text line position determination in digitized documents. *Pattern Recognition*, to appear.

Gonzalez, R. C. and Woods, R. E. (1993) *Digital Image Processing*. Addison-Wesley, Reading, MA.

Impedovo, S., Ottaviano, L. and Occhinegro, S. (1991) Optical character recognition. A survey. *International Journal of Pattern Recognition and Artificial Intelligence*, **5**, 1–23.

Jain, A. K. (1989) *Fundamentals of Digital Image Processing*. Prentice-Hall, Englewood Cliffs, NJ.

Kahan, S., Pavlidis, T. and Baird, H. S. (1987) On the recognition of printed characters of any font and size. *IEEE Trans. Pattern Anal. Mach. Intell.*, **9**, 274–287.

Kerrick, D. and Bovik, A. (1988) Microprocessor-based recognition of handprinted characters from a tablet input. *Pattern Recognition*, **21**(5), 525–537.

Khotanzad, A. and Hong, Y. H. (1990) Invariant image recognition by Zernike moments. *IEEE Trans. Pattern Anal. Mach. Intell.*, **12**(5), 489–497.

Kovacs, Z. S. and Guerrieri, R. (1995) Massively-parallel handwritten character recognition based on the distance transform. *Pattern Recognition*, **28**(3), 293–301.

Papamarkos, N. and Gatos, B. (1994) A new approach for multithreshold selection. Computer vision. *Graphics, and Image Processing—Graphical Models and Image Processing*, **56**(5), 357–370.

Papamarkos, N., Spiliotis, I. and Zoumadakis, T. (1994) Character recognition by signature approximation. *International Journal of Pattern Recognition and Artificial Intelligence*, **8**(5), 1171–1187.

Rogova, G. (1994) Combining the results of several neural network classifiers. *Neural Networks*, **7**(5), 777–781.

Shridhar, M. and Badreldin, A. (1984) High accuracy character recognition algorithm using Fourier and topological descriptors. *Pattern Recognition*, **17**(5), 515–524.

Wahl, F. M., Wong, K. Y. and Casey, R. G. (1982) Block segmentation and text extraction in mixed text/image documents. *Computer Graphics and Image Processing*, **20**, 375–390.

Wang, D. and Shihari, S. N. (1989) Classification of newspaper image blocks using texture analysis. *Computer Vision Graphics and Image Processing*, **47**, 327–352.

Wong, K. Y., Casey, R. G. and Wahl, F. M. (1982) Document analysis system. *IBM J. Res. Devel.*, **26**(6), 647–656.

Xu, L. and Krzyzak, A. (1992) Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Trans. on Systems, Man, and Cybernetics*, **22**(3), 418–435.

AUTHORS' BIOGRAPHIES

Basilios Gatos was born in Athens, Greece, in 1967. He received his Diploma Degree in Electrical Engineering from the Democritus University of Thrace, in 1992. At present, he is working on his Ph.D. Degree in the Electrical and Computer Engineering Department of Democritus University of Thrace, in the field of Optical Character Recognition. Since January 1993, he has been a Postgraduate Scholarship holder of the Institute of Informatics and Telecommunications, National Research Centre "Demokritos". His current interests include document analysis and image processing. He is a member of the Greek Technical Chamber.

Nikos Papamarkos was born in Alexandroupoli, Greece, in 1956. He received his Diploma Degree in Electrical and Mechanical Engineering from the University of Thessaloniki, Greece, in 1979, and a Ph.D. Degree in Electrical Engineering in 1986, from the Democritus University of Thrace, Greece. From 1987 to 1990 Dr Papamarkos was a Lecturer, and from 1990 to 1996 an Assistant Professor at the Democritus University of Thrace, where he is currently Associate Professor (since 1996). His current research interests are in digital signal processing, filter design, image processing, pattern recognition and computer vision. Dr Papamarkos is a member of the IEEE and a member of the Greek Technical Chamber.

Christodoulos Chamzas was born in Komotini, Greece. He received a Diploma Degree in Electrical and Mechanical Engineering from the National Technical University, Athens, Greece, in 1974, and M.S. and Ph.D. degrees in Electrical Engineering in 1975 and 1979 from the Polytechnic Institute of New York. He was an Assistant Professor at the Polytechnic Institute of New York (1979–1982), a member of the Visual Communications Research Department, AT&T Bell Laboratories (1982–1990) and has been an Associate Professor at the Democritus University of Thrace since 1992. His primary interests are in signal processing, image coding, multimedia and communications systems. Dr Chamzas is a member of the Technical Chamber of Greece and Sigma Xi, an Editor of the IEEE Transactions of Communications and a Senior Member of the IEEE.