Processing and Recognition of Handwritten Documents

Georgios Vamvakas *

[1] Department of Informatics and Telecommunications
National and Kapodistrian University of Athens

[2] Computational Intelligence Laboratory
Institute of Informatics and Telecommunications
National Centre for Scientific Research "Demokritos"

gbam@iit.demokritos.gr

Abstract. Nowadays, the accurate recognition of machine printed characters is considered largely a solved problem. A lot of commercial products are focused towards that direction, achieving high recognition rates. However, handwritten character recognition is comparatively difficult. So, the recognition of handwritten documents is still a subject of active research. In this thesis we studied the processing and focused on the recognition stages for handwritten optical character recognition. At the recognition stage a feature vector is extracted for all extracted characters in order to classify them to predefined classes using machine learning techniques. We studied several feature extraction techniques and developed methodologies that efficiently combine different types of features. Furthermore, a novel methodology that extracts features and classifies characters using a hierarchical scheme is proposed. This methodology, after being tested on well-known character databases, as well as on databases consisting of characters from historical documents and a database consisting of Greek contemporary handwritten characters, that were particularly created in this thesis, achieved recognition rates that are among the best one can find in the literature. This methodology was also applied to cursive handwritten words. The recognition rates in these experiments were also very high. Finally, an algorithm that automatically estimates the free parameters involved in character segmentation is also suggested. Character segmentation is very important because its result affects directly the recognition rates. Thus, the optimal segmentation is essential for a successful recognition.

Keywords: handwritten character recognition, feature extraction, hierarchical classification, machine learning techniques, character databases

1 Introduction

The large amount of documents that we have in our possession nowadays, due to the expansion of digital libraries, has pointed out the need for reliable and accurate Optical Character Recognition (OCR) systems for processing them. Although, the accurate recognition of contemporary machine printed characters is considered largely a solved problem, as mentioned above, handwritten character recognition is comparatively difficult, due to different handwriting styles, cursive handwriting and possible skew.

Another challenging task in OCR is the recognition of historical documents. Such documents are of great importance because they are a significant part of our cultural heritage. However, their low quality, the lack of standard alphabets and the presence of unknown fonts are major drawbacks in achieving high recognition

* Dissertation Advisors: [1] Sergios Theodoridis, Professor – [2] Dr. Basilis Gatos, Researcher

rates. In case of historical document processing in particular, an important area is word spotting. In many cases due to high levels of distortion, extremely poor quality, cursive handwriting etc. such documents can not be processed by an OCR system. In order to extract information from these documents page retrieval approaches, for searching or indexing, are adopted. However, this has to be done manually. In essence, this means that each occurrence of a word in a corpus must be annotated by hand. The goal of the word spotting idea is to greatly reduce the amount of annotation work that has to be performed.

According to the above, one can easily realize there are various issues that an OCR system have to deal with, such as character/word recognition and word spotting for either historical or contemporary documents. In this thesis, we suggest novel methodologies that attempt to deal with such issues, thus trying to assist document image processing. Moreover, we also propose an automatic unsupervised free parameter selection approach that optimizes the character segmentation algorithm adopted. This is essential because the segmentation step affects directly the recognition result.

2 Related Work

A widely used approach in OCR systems is to follow a two step schema: a) represent the image as a vector of features and b) classify the feature vector into classes. Selection of a feature extraction method is important in achieving high recognition performance. A feature extraction algorithm must be robust enough so that for a variety of instances of the same symbol, similar feature sets are generated, thereby making the subsequent classification task less difficult [2].

Feature extraction methods have been based mainly on three types of features [1, 2, 3 and 4]: a) statistical derived from statistical distribution of points b) structural and c) transformation-based or moment-based features. A survey on feature extraction methods can be found in [5]. Moreover, other approaches focus on measuring the similarity/dissimilarity between shapes by mapping one character onto another [6, 7].

All the above feature extraction techniques have been applied with great success to both historical and contemporary document recognition. However, there are also methodologies focused on the unique characteristics of the corresponding historical document they process, such as content and writing style [8, 9].

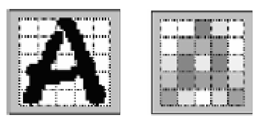
There have been quite a number of successes in determination of invariant features and a wide range of classification methods have been extensively researched. However, as mentioned in [10], most character recognition techniques use a "one model fits all" approach, i.e. a set of features and a classification method are developed and every test pattern is subjected to the same process regardless of the constraints present in the problem domain. It is shown that approaches which employ a hierarchical treatment of patterns can have considerable advantages compared to the "one model fits all" approaches, not only improving the recognition accuracy but also reducing the computational cost as well.

Most classification strategies in OCR deal with a large number of classes trying to find the best discrimination among them. However, such approaches are vulnerable to classification errors when patterns of similar shapes are present since they are not easily distinguished. In [11] a two-stage classification approach is presented to detect and solve possible conflicts between patterns with similar shapes. During the first stage, a single classifier or ensemble of classifiers detect potential conflicts. The second processing stage becomes active only when a decision on the difficult cases must be taken. A comparative study between three different two-stage hierarchical learning architectures can be found in [12].

Word-spotting techniques for searching and indexing historical documents have been introduced. In [13], word images are grouped into clusters of similar words by using image matching to find similarity. Then, by annotating "interesting" clusters, an index that links words to the locations where they occur can be built automatically. In [14] and [15] holistic word recognition approaches for historical documents are presented. Their goal is to produce reasonable recognition accuracies which enable performing retrieval of handwritten pages from a user-supplied ASCII query.

3 Efficient Combination of Feature Extraction Techniques

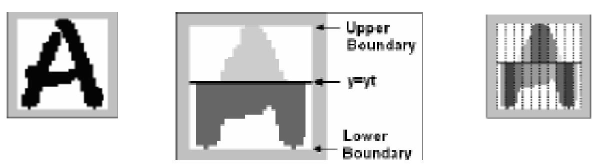
In our approach [16], we employ four types of features. The first set of features is based on zones. The image is divided into horizontal and vertical zones, and for each zone we calculate the density of the character pixels (Fig. 1).



(a) (b)

Figure 1. Feature extraction of a character image based on zones. (a) The normalized character image. (b) Features based on zones. Darker squares indicate higher density of character pixels.

In the second type of features, the area that is formed from the projections of the upper and lower as well as of the left and right character profiles is calculated. Firstly, the center mass (x_t,y_t) of the character image is found.



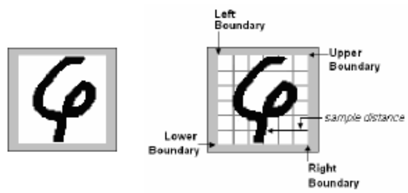
(a) (b) (c)

Figure 2. Feature extraction of a character image based on upper and lower character profile projections. (a) The normalized character image. (b) Upper and lower character profiles. (c) The extracted features. Darker squares indicate higher density of zone pixels.

Upper/lower profiles are computed by considering, for each image column, the distance between the horizontal line $y=y_t$ and the closest pixel to the upper/lower boundary of the character image (Fig. 2b). This ends up in two zones (upper, lower) depending on y_t. Then both zones are divided into vertical blocks. For all blocks formed we calculate the area of the upper/lower character profiles. Fig. 2c illustrates the features extracted from a character image using upper/lower character profiles. Similarly, we extract the features based on left/right character profiles.

The third feature set is based on the distances of the first image pixel detected from the upper and lower boundaries of the image, scanning along equally spaced vertical lines as well as from the left and right boundaries scanning along equally spaced horizontal lines (Fig .3).

The forth set, calculates the profiles of the character from the upper, lower, left and right boundaries of the image, as shown in Fig. 4. The profile counts the number of pixels between the edges of the image and the contour of the character. These features are used because they describe well the external shape of the characters.



(a) (b)

Figure 3. Feature extraction of a character image based on distances. (a) The normalized character image. (b) A sample distance from the right boundary.

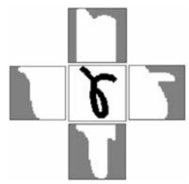


Figure 4. Features extraction of the character image based

Our methodology [16] for character recognition also considered a dimensionality reduction step, according to which the dimension of the feature space, engendered by the features extracted as described above, is lowered down to comprise only the features pertinent to the discrimination of characters into the given set of letters. In particular, we employed the Linear Discriminant Analysis (LDA) method, according to which the most significant linear features are those where the samples distribution has important overall variance while the samples per class distributions have small variance. Formally, this criterion is represented:

$$\mathrm{LDA}(w) = \frac{w^T \operatorname{Cov}(X) w}{w^T \mathrm{E}_c\left[\operatorname{Cov}(X \mid c)\right] w} \tag{1}$$

where w represents a linear combination of the original features, X the original feature vector, c the class, Cov is a the covariance matrix that has to be estimated from the samples and E_c is the expectation in respect to the classes. It turns out that finding the linear features that maximize the LDA criterion comes down to solving a generalized eigenvalue/eigenvector problem and keeping the eigenvectors that have greater eigenvalues. Moreover, the ratio of the sum of the eigenvalues kept to the overall eigenvalues sum provides as an index of quality of the feature subspace kept.

4 Hierarchical Character/Word Recognition

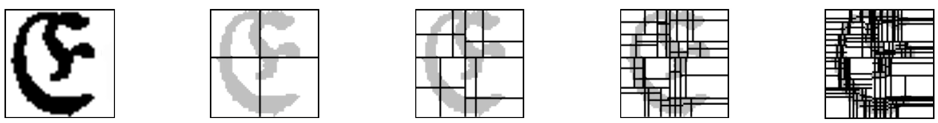
In this section a new feature extraction method followed by a hierarchical classification scheme is presented [17].

4.1 Feature Extraction

4.1.1 Characters

Let $im(x,y)$ be the character image array having 1s for foreground and 0s for background pixels and x_{max} and y_{max} be the width and the height of the character image. Our feature extraction method relies on iterative subdivisions of the character image, so that the resulting sub-images at each iteration have balanced (approximately equal) numbers of foreground pixels, as far as this is possible. At the first iteration step (zero level of granularity, that is $L = 0$) the character image is subdivided into four rectangular sub-images using a vertical and a horizontal divider line as follows: Firstly, a vertical line is drawn that minimizes the absolute difference of the number of foreground pixels in the two sub-images to its left and to its right. Subsequently, a horizontal line is drawn that minimizes the absolute difference of the number of the foreground pixels in the two sub-images above and below. An important point is that the above dividing lines are determined taking into account sub-pixel accuracy. The pixel at the intersection of the two lines is referred to as the *division point* (DP). At further iteration steps (levels of granularity L=1, 2, 3 …), each sub-image obtained at the previous step is divided into four further sub-images using the same procedure as above (Fig.5).

Let L be the current level of granularity. At this level the number of the sub-images is $4^{(L+1)}$. For example, when $L = 0$ (Fig.5b) the number of sub-images is 4 and when $L = 1$ it is 16 (Fig.5c). The number of DPs at level L equals to 4^L. At level L, the co-ordinates (x_i, y_i) of all DPs are stored as features. So, for every L a $2*4^L$ - dimensional feature vector is extracted.



(a) (b) (c) (d) (e)

Figure 5. Character image and sub-images based on DP: (a) original image, (b), (c), (d), (e) subdivisions at levels 0, 1, 2 and 3 respectively.

After all feature vectors are extracted each feature is scaled to [0, 1]. Since each character is normalized to an NxN matrix all feature values f are in the range of [1, N]. Therefore, the value f_i of the i_{th} feature of every feature vector is normalized according to Eq.2.

$$f_i^{'} = \frac{f_i}{N} \tag{2}$$

4.1.2 Words

In case of word recognition [18] the feature extraction technique applied for characters is also adopted. However, in order for features to be invariant of scaling the feature vector does not consist of the co-ordinates (x_i, y_i) of all DPs at a level L but of the pairs ($x_i - x_0$, $y_i - y_0$), where x_i, y_i are the co-ordinates of the DP at L and x_0, y_0 are the co-ordinates of the initial DP (at level $L = 0$) of the word image. Furthermore, all features are normalized in the range of [-1, 1]. Since every word is normalized to an NxM matrix all feature values are scaled according to Eq. 3 and 4.

$$x_i^{'} = \frac{x_i - x_0}{N} \tag{3}$$

$$y_i^{'} = \frac{y_i - y_0}{M} \tag{4}$$

4.2 Hierarchical Classification

For the recognition procedure a hierarchical classification scheme is employed. Since characters/words with similar structure are often mutually confused when using a certain granularity feature representation, we propose to merge the corresponding classes at this level of classification. At a next step, we distinguish those character/word classes by employing a feature vector extracted at another level of granularity where the misclassifications between them are the least possible. The proposed classification scheme has a) a training and b) a recognition phase:

a. Training Phase

The training phase consists of three distinct steps: Step 1 is used to determine the level with the highest recognition rate for the initial classification, step 2 to merge mutually misclassified classes at the level found in step 1 and step 3 to find the level at which each group of merged classes is distinguished the best and to train a new classifier for each one at this level. These steps are described below:

Step 1: Starting from level 1 and gradually proceeding to higher levels of granularity, features are extracted, the confusion matrix is created and the overall recognition rate is calculated, until the recognition rate stops increasing The level at which the highest recognition rate is achieved is considered to be the best performing granularity level (*BPGL*) . Confusion matrices are created at each level from the training set using a *K*-fold cross-validation process. In our case *K* is set to 10.

Step 2: At *BPGL* where the maximum recognition rate is obtained the corresponding confusion matrix is scanned and classes with high misclassification rates are merged. Class merging is performed using the *disjoint grouping scheme* presented in [12]. Let the confusion matrix for C classes be $A_{i,j}$, where $A_{i,j}$ (i, j = 1, 2 … C) is the number of samples that belong to class i and are classified to class j. The similarity between classes i and j is defined according to Eq. 5.

$$N_{i,j} = A_{i,j} + A_{j,i}, (i < j) \tag{5}$$

Suppose we have two groups of classes G_p and G_q having m and n classes respectively. The similarity between these groups ($p < q$) is defined as:

$$S_{p,q} = \min_{i<j} N_{i,j}, (i = i_1 ... i_m, j = j_1 ... j_n) \tag{6}$$

Initially each class is a group. First two classes i and j with the highest $N_{i,j}$ value are found and merged into one group thus resulting in $C - 1$ groups. Next, the most similar groups according to Eq. 6 are merged into one. The procedure is iterated until all similarity values between groups are equal to zero in order to find all possible misclassifications.

Step 3: For each group of classes found in Step 2 the procedure described in Step 1 is performed again and the best distinguishing granularity level (*BDGL*) for its classes is found. Then, for every group another classifier is trained with features extracted at its *BDGL* in order to distinguish the merged classes at the next stage of the classification.

b. Recognition Phase

Each pattern of the test set is fed to the initial classifier with features extracted at *BPGL*. If the classifier decides that this pattern belongs to one of the non-group classes then its decision is taken into consideration and the unknown pattern is assumed to be classified. Else, if it is classified to one of the group classes then it is given to the group's corresponding classifier and this new classifier decides about the recognition result. Note that if a sample is wrongly classified to a non-group class then at the next stage it will remain wrong. However, if it is misclassified to a group-class then it is possible to be correctly classified in the second stage.

5 Word Spotting

In this section a word spotting technique, based on the combination of results from different levels of the feature extraction method described in Section 4, is introduced [18]. Given a keyword that we want to match in a set of document images that have been segmented at word level, the matching algorithm is applied as follows:

Step 1: Create five lists R_i, i = 1, 2, 3, 4 and 5, each one consisting of the *Euclidean Distances* between the keyword and every word of the set of documents, using feature vectors from granularity levels L_i, i = 1, 2, 3, 4 and 5 respectively, extracted according to the procedure described in Section 3.2.

Step 2: Normalize all distances in each R_i to [0, 1] by diving each one with the maximum distance in R_i.

Step 3: Merge all five lists in a list Q. Every word of the set of documents is represented in Q by five distances from the keyword. For each one we choose to keep the minimum distance and remove the others, resulting to Q'.

Step 4: Sort Q' in ascending order. Choose a threshold *thr* and keep only the first *thr* instances. List Q' now contains only the *thr* nearest words to the word we want to be matched.

6 Automatic Unsupervised Parameter Selection for Character Segmentation

Character segmentation is a difficult problem since low quality of document images and the wide variety of fonts can cause touching and broken characters. In most segmentation approaches a major problem is the selection of the free parameters that affect directly the segmentation results. The parameters are either user-specified and no training method is included [19, 20 and 21] or selected through a training procedure over a set of “optimal” parameter values that are usually manually selected based on some assumption regarding the training data [22]. However, ground truth or a priori knowledge of the fonts of the document image is not always available. To this end, we introduce a novel automatic unsupervised parameter selection methodology for character segmentation that is based on clustering [23]. The clustering is performed using features extracted from the segmented entities based on zones and from the area that is formed from the projections of the upper/lower and left/right profiles as described in Section 3. Optimization of an appropriate intra-class distance measure yields the optimal parameter vector.

Consider a character segmentation algorithm whose result depends on P parameters. Let S_1, S_2 … S_v different parameter vectors (p-tuples) for different values of the parameters obtained using a standard selection method (e.g. random selection, selection through a grid). In our approach the well-known k-Means clustering algorithm is adopted due to its computational simplicity and the fact that, as all clustering techniques which use point representatives, is suitable for recovering compact clusters. If the expected number of different characters is in the interval between k_1 and k_2, then for every S_q we proceed to a k-Means clustering with k taking values from k_1 to k_2.

Given a parameter vector S_q, in order to evaluate the performance of the clustering algorithm for every k between k_1 and k_2, the mean squared distances from the centroids (within clusters sum of squares) is calculated as follows:

$$W_q(k) = \sum_{j=1,2..k} \frac{1}{n_{c_j}} \sum_{i \in C_j} d^2(x_i, \overline{x_j}) \tag{7}$$

where $\overline{x_j}$ is the centroid of the cluster C_j, $j = 1,2 \ldots k$, x_i is the i_{th} pattern inside cluster C_j, n_{c_j} is the cardinality of cluster C_j and d is the Euclidean Distance.

The value of $W_q(k)$ is low when the partition is good thus resulting to compact clusters. A measure of the quality of the segmentation result that corresponds to a parameter vector S_q is given as:

$$Q(S_q) = \frac{10^5}{\min\limits_{k=k_1,\ldots,k_2} (W_q(k))} \tag{8}$$

The optimal parameter vector S_{opt} is defined as:

$$S_{opt} = \operatorname*{arg\,max}_{S_q = S_1, S_2, \ldots, S_v} (Q(S_q)) \tag{9}$$

7 Experimental Results

For our experiments the well-known CEDAR CD-ROM-1 [24], a database consisting of Greek handwritten characters (CIL-Database) [16], two databases [26] comprising samples of characters from old Greek Christian documents of the 17th century (HW and TW Databases) and a character database (TW-1 Database) [18] created by a part of a historical book from Eckartshausen which was published on 1788 and is owned by the Bavarian State Library [25]. The HW, TW and TW-1 databases were created using a semi-automatic procedure that relies on clustering as presented in [26]. For word recognition the IAM v3.0 database [29] was employed, as described in [39]. Finally, in order to demonstrate the results of the word spotting algorithm a set of historical handwritten images from George Washington's collection from the Library of Congress [27] was used.

Regarding the classification step, the Support Vector Machines (SVM) algorithm was adopted in conjunction with the Radial Basis Function (RBF) kernel [28].

In case of character recognition Tables 1, 2, 3, 4 and 5 show the experimental results for CIL, HW, TW, TW-1 and CEDAR databases, while Table 6 depicts the recognition accuracy for word recognition using the IAM database.

Table 1. Recognition Rates for CIL Database

CIL Database	
Zones	88.48%
Projections	87.75%
Distances	82.53%
Profiles	83.25%
Zones + Projections [30]	91.68%
Zones + Projections + Distances + Profiles with LDA [16]	92.05%
Hierarchical Classification v.1 [31]	93.21%
Hierarchical Classification v.3 [17]	**95.63%**

Table 2. Recognition rates for HW,TW and TW-1 Databases

	HW	TW	TW-1
Zones + Projections [23]	94.62%	95.44%	NA
Hierarchical Classification v.1 [32]	94.51%	97.71%	NA
Hierarchical Classification v.3 [17]	**95.21%**	**98.24%**	**99.53 %**

Table 3. Recognition rates for the CEDAR Database (52 Classes, a-z/A-Z)

CEDAR Database			
	Uppercase Characters	**Lowercase Characters**	**Overall Recognition Rate**
YAM[33]	NA	NA	75.70%
KIM [34]	NA	NA	73.25%
GAD[35]	79.23%	70.31%	74.77%
Hierarchical Classification v.3 [17]	**86.17%**	**84.05%**	**85.11%**

.

Table 4. Recognition rates for the CEDAR Database for uppercase only and lowercase only characters.

CEDAR Database						
	Uppercase Characters (26 Classes)			**Lowercase Characters (26 Classes)**		
	# Train Patterns	**# Test Patterns**	**Recognition Rate**	**# Train Patterns**	**# Test Patterns**	**Recognition Rate**
BLU[36]	7175	939	81.58%	18655	2240	71.52%
Hierarchical Classification v.3 [17]	11454	1367	**95.90%**	7691	816	**93.50%**

Table 5. Recognition rates for the CEDAR Database after merging lowercase and uppercase characters with similar shapes.

CEDAR Database				
	Number of Classes (all classes)	**Recognition Rate**	**Number of Classes (after merging)**	**Recognition Rate**
SIN [37]	52	NA	36	67%
CAM [38]	52	83.74%	39	84.52%
Hierarchical Classification v.3 [17]	52	**85.11%**	35	**94.73%**

Table 6. Recognition rates for the IAM Database.

IAM Database	
GAT [39]	87.68%
Hierarchical Classification v.3 (for words) [18]	**90.56%**

Regarding the experiments for word spotting two datasets from [25] (dataset-1) and from [27] (dataset-2) were used, for evaluating the proposed feature extraction technique, consisting of 13 and 10 document images respectively. Moreover, three words from each dataset were used as keywords: “Durchleucht”, “nicht” and “Natur” that appear 10, 21, and 17 times respectively in dataset-1 and “public”, “appointments” and “government” that appear 9, 10 and 8 times respectively in dataset-2. Tables 7 and 8 present the *F*-measure rates using different values of threshold *thr*.

Table 7. *F*-Measure for dataset-1

Keyword	**Threshold (*thr*)**					
	5	**10**	**15**	**20**	**25**	**30**
Durchleucht	66.67%	50%	40%	40%	40%	40%
nicht	38.45%	64.50%	83.32%	97.55%	86.95%	82.35%
Natur	45.45%	74.07%	67.28%	70.27%	71.42%	72.38%

Table 8. *F*-Measure for dataset-2

Keyword	**Threshold (*thr*)**					
	5	**10**	**15**	**20**	**25**	**30**
appointments	75%	77.77%	80%	72.72%	74.99%	69.23%
public	80%	82.34%	73.67%	76.19%	69.56%	72%
government	42.85%	37.5%	33.34%	30%	36.36%	35.71%

8 Concluding Remarks

In this thesis novel methodologies that assist handwritten and historical document recognition are presented. In particular: An efficient feature extraction using different types of features followed by a dimensionality reduction step is proposed. Moreover, a novel feature extraction based on recursive subdivisions of the image is introduced. Even though the feature extraction method itself is quite efficient when a specific level of granularity is used, there is more to be gained in classification accuracy by exploiting the intrinsically recursive nature of the method. This is achieved by appropriately combining the results from different levels using a hierarchical approach. Several databases, historical or contemporary, were used to evaluate the performance of these methodologies. In all cases the experimentations depicted, regarding other state-of–the-art techniques that the recognition rates either for characters or words are among the highest one can find in the literature. Also, a new word-spotting algorithm is suggested that relies on the combination of features extracted at different levels of granularity. Finally, a methodology for automatic unsupervised parameter selection for character segmentation is proposed. The methodology is based on clustering; suggesting that the optimal segmentation output, relying on a set of parameters, should produce the best clustering. Experimental results, based on evaluation of segmentation using the ground truth, show that the proposed methodology is capable of finding the optimal or near optimal parameter set.

Figure 6 shows the recent achievements is character recognition. It is obvious that handwritten character recognition as well as historical character recognition that consist of symbols or ligatures that no longer exist in modern alphabets, are still active in research.

		Machine Printed			Handwritten		
		Single Font	Omni Font	Multi Font	Discrete	Cursive	Mixed
On-Line	Constrained				2	3	3
	Unconstrained				1	1	1
Off-Line	Noiseless	2	2	2	3	1	1
	Noisy	1	1	1	1	1	1

1 Need more research   2 Needs improvement   3 Well Done

Figure 6. Recent achievements in OCR

References

1. Luiz S. Oliveira, F. Bortolozzi, C.Y.Suen, "Automatic Recognition of Handwritten Numerical Strings: A Recognition and Verification Strategy", IEEE Transactions on Pattern Recognition and Machine Intelligence, 2001, Vol. 24, No. 11, pp. 1448-1456.
2. K. M. Mohiuddin and J. Mao, "A Comprehensive Study of Different Classifiers for Hand-printed Character Recognition", Pattern Recognition, Practice IV, 1994, pp. 437- 448.
3. L. A. Koerich, "Unconstrained Handwritten Character Recognition Using Different Classification Strategies", International Workshop on Artificial Neural Networks in Pattern Recognition (ANNPR), 2003.
4. N. Arica and F. Yarman-Vural, "An Overview of Character Recognition Focused on Off-line Handwriting", IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2001, 31(2), pp. 216 - 233.
5. O. D. Trier, A. K. Jain, T.Taxt, "Features Extraction Methods for Character Recognition – A Survey ", Pattern Recognition, 1996, Vol.29, No.4, pp. 641-662.
6. S. Belongie, J. Malik, J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.24, No. 4, pp. 509-522, 2002.
7. Anil K. Jain, Dougla Zongker, "Representation and Recognition of Handwritten Digits using Deformable Templates", IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, Vol. 19, No. 12, pp. 1386-1391.
8. V.G.Gezerlis and S.Theodoridis, "Optical Character Recognition for the Orthodox Hellenic Byzantine music notation", Pattern Recognition, 2002, Vol.35, pp. 895 – 914.
9. K. Ntzios, B. Gatos, I. Pratikakis, T. Konidaris and S.J. Perantonis, "An Old Greek Handwritten OCR System based on an Efficient Segmentation-free Approach", International Journal on Document Analysis and Recognition (IJDAR), Special Issue on Historical Documents, 2007,vVol. 9, No. 2-4, pp. 179-192.
10. J. Park, V. Govindaraju, S. N. Shrihari, "OCR in Hierarchical Feature Space", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, Vol. 22, No. 24, pp. 400-408.
11. L. Vuurpijl, L. Schomacker amd M. van Erp, "Architecture for detecting and solving conflicts: two-stage classification and support classifiers", International Journal on Document Analysis and Recognition (IJDAR), 2004. Vol. 5, No. 4, pp.213-223.
12. Tapan Kumar Bhowmik, Pradip Ghanty, Anandarup Roy and Swapan Kumar Parui, "SVM-Based Hierarchical Architectures for Handwritten Bangla Character Recognition" , Int. Journal of Document Analysis and Recognition (IJDAR), 2009, 12 (2), pp. 97-108.
13. T.M.Rath and R. Manmatha, "Word spotting for historical documents", International Journal on Document Analysis and Recognition (IJDAR), 2006, Vol.9, No 2 – 4, pp. 139 – 152.
14. V. Lavrenko, T. M. Rath, R. Manmatha: "Holistic Word Recognition for Handwritten Historical Documents", Proceedings of the First International Workshop on Document Image Analysis for Libraries (DIAL'04), 2004, pp 278-287.
15. T. Adamek, N. E. O'Connor, A. F. Smeaton, "Word Matching Using Single-Closed Contours for Indexing Handwritten Historical Documents", International Journal on Document Analysis and Recognition (IJDAR), Special Issue on Analysis of Historical Documents, 2006.
16. G. Vamvakas, B. Gatos, S. Petridis and N. Stamatopoulos, "An Efficient Feature Extraction and Dimensionality Reduction Scheme for Isolated Greek Handwritten Character Recognition", Proceedings of the 9^{th} International Conference on Document Analysis and Recognition, Curitiba, Brazil, 2007, pp. 1073-1077.

17. G. Vamvakas, B. Gatos, S. J. Perantonis, “Handwritten Character Recognition through Two-Stage Foreground Sub-Sampling”, Pattern Recognition, Vol.43, Issue 8, pp. 2807-2816, 2010.
18. G. Vamvakas, B. Gatos, S. J. Perantonis, “Efficient Character/ Word Recognition based on a Hierarchical Classification Scheme”, International Journal on Document Analysis and Recognition (IJDAR), suggested for the Special Issue: ICDAR’O9, to appear.
19. Antonacopoulos, A., Karatzas, D., “Semantics-based content extraction in typewritten historical documents”, in: Eighth International Conference on Document Analysis and Recognition, 48–53, 2005.
20. Liang, S., Shridhar, M., and Ahmadi, M.. “Segmentation of touching characters in printed document recognition”, Pattern Recognition 27 (6), 825–840, 1994.
21. Nikolaou, N., Makridis, M., Gatos, B., Stamatopoulos N., and Papamarkos, N., “Segmentation of historical machine-printed documents using Adaptive Run Length Smoothing and skeleton segmentation paths”, Image and Vision Computing, doi:10.1016/j.imavis.2009.09.013, 2009.
22. Kavallieratou E., Stamatatos E., Fakotakis N., Kokkinakis G., “Handwritten character segmentation using transformation-based learning”, 15th International Conference on Pattern Recognition, vol. 2, pp. 634–637, 2000.
23. G.Vamvakas, N. Stamatopoulos, B.Gatos, S.J.Perantonis, “Automatic Unsupervised Parameter Selection for Character Segmentation” , 9th IAPR International Workshop on Document Analysis Systems (DAS’10), pp 409-416, June 9-11, Boston, USA, 2010.
24. J.J. Hull, “A database for handwritten text recognition research”, IEEE Trans. Pattern Anal. Mach. Intell. 16 (5) (1994) 550–554.
25. Carl von Eckartshausen, 1778, Aufschlüsse zur Magie aus geprüften Erfahrungen über verborgene philosophische Wissenschaften und verdeckte Geheimnisse der Natur”, Bavarian State Library.
26. G. Vamvakas, B. Gatos, N. Stamatopoulos, S.J. Perantonis, "A Complete Optical Character Recognition Methodology for Historical Documents," 8th IAPR International Workshop on Document Analysis Systems (DAS’08), pp.525-532, Nara, Japan, September 2008.
27. http://memory.loc.gov/ammem/gwhtml/gwhome.html
28. Cortes C., and Vapnik, V, " Support-vector network ", Machine Learning, vol. 20, pp. 273-297, 1997.
29. IAM Handwritten Database v3.0 , http://www.iam.unibe.ch/~fki/iamDB/.
30. G. Vamvakas, B. Gatos, I. Pratikakis, N. Stamatopoulos, A. Roniotis, S.J. Perantonis, "Hybrid Off-Line OCR for Isolated Handwritten Greek Characters", 4th IASTED International Conference on Signal Processing, Pattern Recognition and Applications (SPPRA’07), pp. 197-202, Innsbruck, Austria,2007.
31. G. Vamvakas, B. Gatos, S. J. Perantonis, "Hierarchical Classification of Handwritten Characters based on Novel Structural Features" (ICFHR'08), 11th International Conference on Frontiers in Handwriting Recognition,Montreal, Canada, August 2008.
32. G. Vamvakas, B. Gatos, S. J. Perantonis, “A Novel Feature Extraction and Classification Methodology for the Recognition of Historical Documents”, 10th International Conference on Document Analysis and Recognition (ICDAR’09), pp 491-495, Barcelona, Spain, July 2009.
33. H. Yamada and Y. Nakano, "Cursive Handwritten Word Recognition Using Multiple Segmentation Determined by Contour Analysis", IECIE Transactions on Information and System, Vol. E79-D. pp. 464-470, 1996.
34. F. Kimura. N. Kayahara. Y. Miyake and M. Shridhar, "Machine and Human Recognition of Segmented Characters from Handwritten Words", International Conference on Document Analysis and Recognition (ICDAR '97), Ulm, Germany, 1997, pp. 866-869.
35. P. D. Gader, M. Mohamed and J-H. Chiang. "Handwritten Word Recognition with Character and Inter-Character Neural Networks", IEEE Transactions on System, Man. and Cybernetics-Part B: Cybernetics, Vol. 27, 1997, pp. 158-164.
36. M. Blumenstein, X.Y. Liu, B. Verma, "A modified direction feature for cursive character recognition", IEEE International Joint Conference on Neural Networks, 2007, Vol.4, pp. 2983 – 2987.
37. S. Singh and M. Hewitt, "Cursive Digit and Character Recognition on Cedar Database". International Conference on Pattern Recognition, (ICPR 2000), Barcelona, Spain. 2000, pp. 569-572.
38. F. Camastra and A. Vinciarelli. "Combining Neural Gas and Learning Vector Quantization for Cursive Character Recognition", Neurocomputing. vol. 51. 2003, pp. 147-159.
39. B. Gatos, I. Pratikakis, A.L. Kesidis and S.J. Perantonis, "Efficient Off-Line Cursive Handwritten Word Recognition", 10th International Workshop on Frontiers in Handwriting Recognition (IWFHR 2006), La Baule, France, October 2006, pp. 121-125.