Pattern Anal Applic (2006) 8: 305–320
DOI 10.1007/s10044-005-0013-7

THEORETICAL ADVANCES

**B. Gatos · K. Ntzios · I. Pratikakis · S. Petridis
T. Konidaris · S. J. Perantonis**

An efficient segmentation-free approach to assist old Greek handwritten manuscript OCR

Received: 10 April 2004 / Accepted: 24 August 2005 / Published online: 8 November 2005

Abstract Recognition of old Greek manuscripts is essential for quick and efficient content exploitation of the valuable old Greek historical collections. In this paper, we focus on the problem of recognizing early Christian Greek manuscripts written in lower case letters. Based on the existence of closed cavity regions in the majority of characters and character ligatures in these scripts, we propose a novel, segmentation-free, fast and efficient technique that assists the recognition procedure by tracing and recognizing the most frequently appearing characters or character ligatures. First, we detect closed cavities that exist in the character body. Then, the protrusions in the outer contour outline of the connected components that contain the character closed cavities are used for the classification of the area around closed cavities to a specific character or a character ligature. The proposed method gives highly accurate results and offers great assistance to old Greek handwritten manuscript OCR. We also provide additional OCR applications that not only prove the robustness of the proposed method but also demonstrate its generic flavor in case segmentation and text location tasks are very difficult to perform.

Keywords Handwriting recognition · Character recognition · Segmentation-free · Feature extraction · Historical document recognition · Old manuscript recognition

B. Gatos (✉) · K. Ntzios · I. Pratikakis · S. Petridis
T. Konidaris · S. J. Perantonis
Computational Intelligence Laboratory, Institute of Informatics and Telecommunications, National Research Center "Demokritos", 153 10 Athens, Greece
E-mail: bgat@iit.demokritos.gr
URL://www.iit.demokritos.gr/cil
E-mail: ntzios@iit.demokritos.gr
E-mail: ipratika@iit.demokritos.gr
E-mail: petridis@iit.demokritos.gr
E-mail: tkonid@iit.demokritos.gr
E-mail: sper@iit.demokritos.gr

Originality and contribution

In this paper, we present a novel methodology that assists old Greek handwritten manuscript OCR contributing to the cultural heritage conservation. We focus on early Christian Greek manuscripts written in lower case letters. Traditional techniques for handwritten recognition cannot be applied in a straight forward manner to our case, since early Christian Greek manuscripts explore several unique characteristics, such as continuous connection between characters, even between different words. Global or segmentation approaches for handwritten character recognition cannot be applied to our case, since individual words or characters cannot be detected. Several segmentation-free approaches that do not involve any segmentation task are based on significant geometric features, such as short line segments, enclosed regions and corners. Instead, we propose a novel method whose originality is based on two aspects. First, a novel segmentation-free approach based on the detection of the closed cavities, is proposed. This aids in the proposed character representation since the closed cavities exist in the majority of characters and character ligatures. Second, novel features are used that are based on the protrusions in the outer contour of the connected components that contain closed cavities. Experimental results show that the proposed method gives highly accurate results and offers great assistance to old Greek handwritten manuscript interpretation. Furthermore, we also provide additional OCR applications that not only prove the robustness of the proposed method but also demonstrate its generic flavor in case that segmentation and text location tasks are very difficult to perform.

1 Introduction

Recognition of old Greek manuscripts is essential for quick and efficient content exploitation of the valuable old Greek historical collections. In this paper, we focus on early Christian Greek manuscripts written in lower case letters. Specifically, our principal concern constitutes the Sinaitic Codex Number Three, which contains the Book of Job, one of the best Greek manuscripts and one of the major masterpieces of world literature (see Fig. 1a). Written in Hebrew initially, the Book was translated into Greek approximately in the 3rd century BC for the sake of the Hellenized Hebrews of Alexandria.

In the field of handwritten character recognition great progress has occurred during the past years [1]. Many methods were developed for a variety of applications like automatic reading of postal addresses [2, 3], fax forms [4] and bank checks [5, 6], form processing, etc. In the literature, two main approaches can be identified: the global approach [7, 8] and the segmentation approach [9, 10]. The global approach entails the recognition of the whole word while the segmentation approach requires that each word has to be segmented into letters.

In the segmentation approach, the crucial step is to split a scanned bitmap image of a document into individual characters. Many segmentation algorithms have been proposed for handwritten words and digits. Lu and Shridhar have reviewed various techniques for the segmentation of handwritten characters [11]. Xiao and Leedman proposed a segmentation method based on certain knowledge of the handwriting [12] while Plamondon and Privitera introduced a segmentation method that partly simulates the cognitive-behavioral process used by human beings in order to recover the temporal sequence of the strokes that composed the original pen movement [13]. Chi et al. proposed a contour curvature-based algorithm to segment single- and double-touching handwritten digits strings [14]. Zhao et al. proposed a two-stage approach to segment unconstrained handwritten Chinese characters [15]. In their algorithm, a character string is first coarsely segmented on the basis of the background skeleton, a vertical projection and a set of geometric features. All possible segmentations paths are evaluated by using the fuzzy decision rules learned from examples discarding unsuitable segmentation paths.

Global approaches avoid character segmentation, looking at words as entities using statistical methods to classify word samples [16]. Holistic strategies employ top–down approaches for recognizing the whole word, thus eliminating the segmentation problem [17–19]. In these strategies, global features extracted from the entire word image are used for the recognition of limited size lexicon. As the size of the lexicon becomes larger, the complexity of algorithms increases linearly due to the need for a larger search space and a more complex pattern representation. Although the global approaches are referred in the literature as "segmentation-free" approaches, they involve a word detection task.

Some approaches that do not involve any segmentation task are based on concepts and techniques that have been used in object recognition with occlusions [20, 21]. According to these approaches, significant geometric features such as short line segments, enclosed regions and corners are extracted from a fully unsegmented raw document bitmap by methods like template matching [22, 23], peephole method [24], *n*-tuple feature [25, 26] and hit-or-miss operator [27].

Traditional techniques for handwriting recognition cannot be applied to early Christian Greek manuscripts written in lower case letters, since continuity between characters of the same or consecutive words does not permit character or word segmentation.

Furthermore, the aforementioned manuscripts entail several unique characteristics as in the following:

- High-script standardization: although, we refer to handwritten manuscripts, the corresponding characters are highly standardized since the manuscripts are immediate predecessors of early printed books.
- Frequent appearance of character ligatures: frequent appearance of closed cavities in the majority of character and character ligatures. As shown in Fig. 1b, closed cavities appear in letters "α", "o", "σ", "ε", "δ", "ω", "π", "θ", "ϕ" as well as in letter ligatures "$\sigma\pi$", "$\varepsilon\sigma$" etc. These constitute 60% of

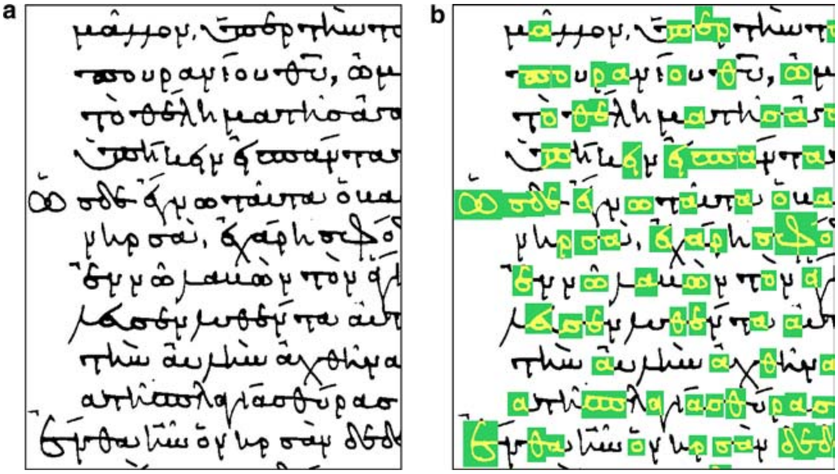


Fig. 1 **a** Early Christian Greek manuscript; **b** Identified characters or character ligatures that contain closed cavities

complete character set used in a typical old Greek manuscript.

The continuity between characters of the same or consecutive words guided us to develop a segmentation-free recognition technique as a fundamental assistance to old Greek handwritten manuscript OCR. Based on the existence of closed cavities in the majority of characters and character ligatures, we propose a technique for the detection and recognition of characters that contain closed cavities. It is a novel method whose originality is based on two aspects. First, a novel segmentation-free approach is used based on the detection of the closed cavities. This aids in the proposed character representation since the hole regions exist in the majority of characters and character ligatures. Second, novel features are used that are based on the protrusions in the outer contour of the connected components that contain closed cavities. In Fig. 2, a flowchart of the proposed handwritten recognition system is shown.

Furthermore, we also provide additional OCR applications that not only prove the robustness of the proposed method but also demonstrate its generic flavor in case segmentation and text location tasks are very difficult to perform.

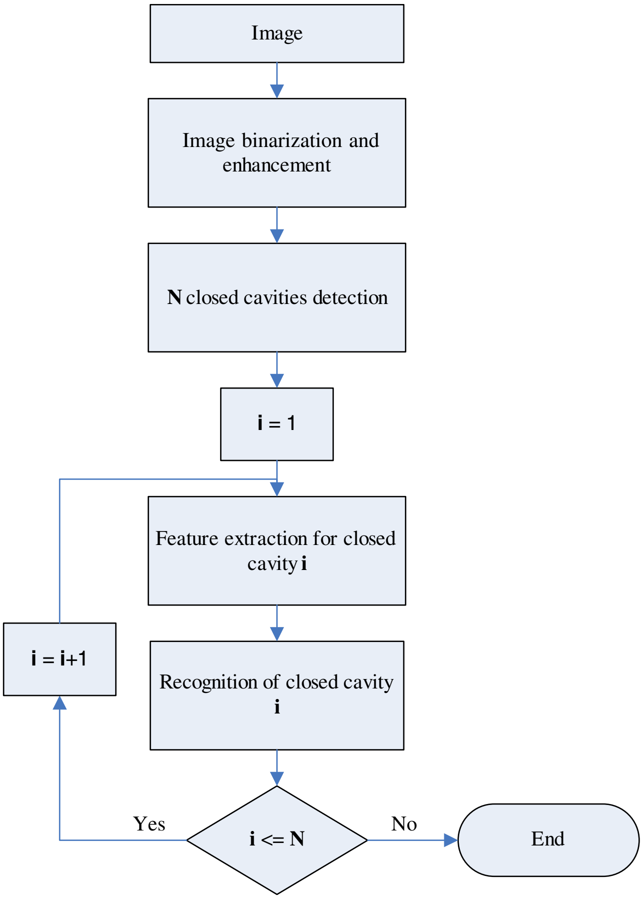


Image
Image binarization and enhancement
N closed cavities detection
i = 1
Feature extraction for closed cavity i
Recognition of closed cavity i
i <= N
Yes
No
End
i = i+1

Fig 2 Flowchart of the handwritten recognition system

The proposed methodology does not aim at a complete character recognition system for old Greek manuscripts. It aims rather at an assessment of the recognition procedure by detecting and recognizing the most frequently appearing characters or character ligatures, using a segmentation-free, quick and efficient approach.

2 Methodology

The proposed methodology consists of several distinct stages. First, we apply a binarization and image enhancement technique to get an improved quality black and white (B/W) image. Second, we trace closed cavities that exist in character bodies. We suggest a novel fast algorithm based on processing the white runs of the initial B/W image. This algorithm permits the extraction of the character-closed cavities but rejects closed cavities of larger dimension, such as closed cavities inside frames, diagrams, etc. In the next step, all closed cavities in characters are initially grouped into several categories based on their spatial proximity and topology. In this way, character closed cavities are classified as: a single closed cavity, two horizontal neighboring closed cavities, three horizontal neighboring closed cavities, four horizontal neighboring closed cavities, two vertical closed cavities and two vertical neighboring patterns that consist of a single closed cavity and two neighboring closed cavities (see Table 1). The final stage of our approach concerns classification of the aforementioned closed cavity patterns into a character or a ligature. It is based on the protrusions that appear in the outer contour outline of the connected components which contain the character closed cavities. The proposed novel recognition methodology and the initially applied binarization and image enhancement tasks are fully described in the following sections.

2.1 Image binarization and enhancement

Binarization is the starting step of most document image analysis systems and refers to the conversion of the gray-scale image to a binary image. Since historical document collections are most of the times of very low quality, an image enhancement stage is also essential. The proposed scheme for image binarization and enhancement is described in [28] and consists of five distinct steps: a preprocessing procedure using a low-pass Wiener filter, a rough estimation of foreground regions using Niblack's approach [29], a background surface calculation by interpolating neighboring background intensities, a thresholding by combining the calculated background surface with the original image and finally a postprocessing step that improves the quality of text regions and preserves stroke connectivity. An example of the image

Table 1 The proposed dictionary for closed cavity patterns

Pattern ID	1	2	3	4	5	6
Pattern	o	o o	o o o	o o o o	o o	o o o
Characters or character ligatures	(α), (o) (ε), (σ) (ρ), (δ)	( ), (ω), (εσ)	 (σ), (ε)	 (α o)	 (θ)	 (φ)

binarization and enhancement result is demonstrated in Fig. 3.

2.2 Character-closed cavity detection

Several closed cavity detection algorithms exist, mainly based on contour following techniques that distinguish the external from internal contours [30, 31]. We suggest a novel fast algorithm for closed cavity detection based on processing the white runs of the B/W image. In the following, a step-by-step description of the proposed algorithm, is given.

Step 1. All horizontal and vertical images white runs that neighbor with image borders or have a length greater than L, get flagged, where L denotes a typical length which reflects character size. The proposed algorithm for closed cavity detection extracts only the character closed cavities and not other closed cavities of larger dimension, with white run length greater than L, such as closed cavities inside frames, diagrams etc.
Step 2. All horizontal and vertical white runs of nonflagged pixels that neighbor with the flagged pixels of step 1, get flagged as well.
Step 3. Repeat step 2 until no pixel remains to be flagged.
Step 4. All remaining white runs of nonflagged pixels belong to image closed cavities. Closed cavities with very small area are ignored.A pseudo-code of the proposed closed cavity detection algorithm is given in Fig. 4, and a demonstration of the above algorithm is shown in Fig. 5.

In early Christian Greek manuscripts, a single character or character ligature may contain more than one closed cavities. Therefore, it is imperative to examine whether the detected closed cavities can be grouped together. This is done by taking into account their spatial proximity and topology leading to the construction of a dictionary. Table 1 shows the proposed dictionary structure. At the last row of this Table, we indicate the corresponding characters and character ligatures.

2.3 Character detection

Feature extraction is applied to characters that contain one or more closed cavities. The proposed method for character isolation creates a bounding box around the

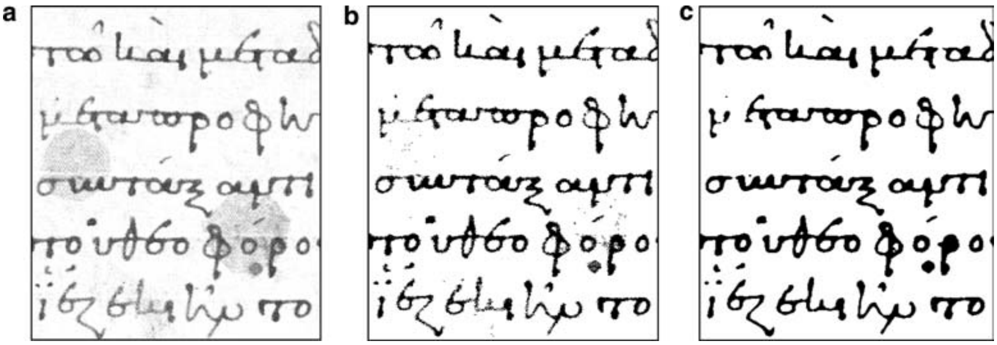


Fig. 3 Image binarization and enhancement example. **a** Original gray scale image; **b** B/W image resulting after binarization; **c** Final image after image enhancement

Fig. 4 Pseudocode of the closed cavity detection algorithm

Closed Cavity Detection Pseudo-code

```
• Document Image is stored in the array IM[x][y], x ∈ [0,$I_x$-1], y ∈ [0,$I_y$-1] which has 1's for the foreground pixels
  ((black) and 0's for the background pixels (white).
• Find [$H_{x1}$(i), $H_y$(i)], [$H_{x2}$(i), $H_y$(i)] which are the coordinates of all horizontal white runs of IM (successive horizontal 0's),
  i ∈ [1,XR].
• Find [$H_x$(j), $H_{y1}$(j)], [$H_x$(j), $H_{y2}$(j)] which are the coordinates of all vertical white runs of IM (successive vertical 0's), j
  ∈ [1,YR].
• For i=1 to XR
  • If ($H_{x2}$(i)- $H_{x1}$(i)>L) OR $H_{x1}$(i)=0 OR $H_{x2}$(i)= $I_x$-1 then
    • For x= $H_{x1}$(i) to $H_{x2}$(i) IM[x][ $H_y$(i)]=2 //FLAG
  • Endif
• Next i
• For j=1 to YR
  • If ($H_{y2}$(j)- $H_{y1}$(j)>L) OR $H_{y1}$(j)=0 OR $H_{y2}$(j)= $I_y$-1 then
    • For y= $H_{y1}$(j) to $H_{y2}$(j) IM[$H_x$(j)][y]=2 //FLAG
  • Endif
• Next j
• Set loop=true
• While loop
  • Set loop=false
  • Recalculate [$H_{x1}$(i), $H_y$(i)], [$H_{x2}$(i), $H_y$(i)] which are the coordinates of all horizontal white runs of IM (successive
    horizontal 0's), i ∈ [1,XR].
  • Recalculate [$H_x$(j), $H_{y1}$(j)], [$H_x$(j), $H_{y2}$(j)] which are the coordinates of all vertical white runs of IM (successive vertical
    0's), j ∈ [1,YR].
  • For i=1 to XR
    • If IM[$H_{x1}$(i)-1][ $H_y$(i)]=2  OR IM[$H_{x2}$(i)+1][ $H_y$(i)]=2 then
      •   Set loop=true
      •   For x=$H_{x1}$(i) to $H_{x2}$(i) IM[x][ $H_y$(i)]=2 //FLAG
    • Endif
  • Next i
  • For j=1 to YR
    • If IM[$H_x$(j)] [$H_{y1}$(j)-1] =2  OR IM[$H_x$(j)] [$H_{y2}$(j)+1]=2 then
      • Set loop=true
      • For y= $H_{y1}$(j) to $H_{y2}$(j) IM[$H_x$(j)][y]=2 //FLAG
    • Endif
  • Next j
• Endwhile
• End of algorithm. All pixels that have IM[x][y]=0 belong to closed cavities.
```

character guided by the spatial relationship between the contours of the closed cavity and the outer contour of the connected component. The coordinates of these contours can be extracted by using a contour following algorithm [31]. Figure 6 shows a part of an image consisting of three connected components each one having one closed cavity.

Let $\mathcal{H} = \left\{\left(x_i^{\mathcal{H}}, y_i^{\mathcal{H}}\right),\ i \in [1,n]\right\}$, be the set of the pixel coordinates, that composes the contour of the closed cavity and $\mathcal{C} = \left\{\left(x_i^{\mathcal{C}}, y_i^{\mathcal{C}}\right),\ i \in [1,m]\right\}$, be the set of the pixel coordinates belonging to the outer contour of the character.

We define the distance between the closed cavity and the connected character component as:

$$D(\mathcal{H},\mathcal{C}) = \max_i\{\min_j\{\sqrt{(x_i^{\mathcal{H}} - x_j^{\mathcal{C}})^2 + (y_i^{\mathcal{H}} - y_j^{\mathcal{C}})^2}\}\} \qquad (1)$$

Calculating $D(\mathcal{H},\mathcal{C})$ and using pixel coordinates which compose the contour of the closed cavity, it is easy to isolate the pixels of the connected character component contained in an adaptive bounding box W with the following top-left (x_{TL},y_{TL}) and bottom right corner coordinates (x_{BR},y_{BR}):

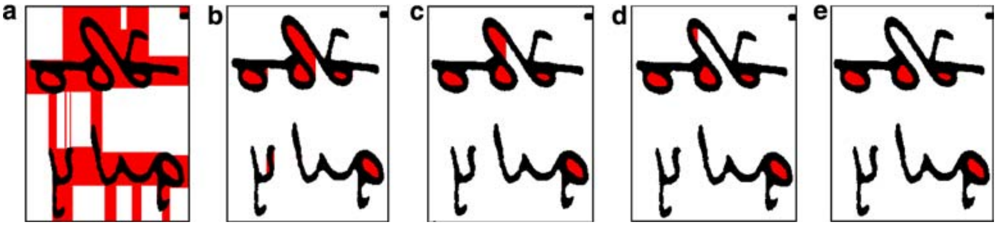


Fig. 5 Demonstration of closed cavity detection algorithm: **a–e** resulting image after 1,2,3,4 and 5 iterations, respectively

$$
\begin{aligned}
x_{TL} &= \begin{cases} \min\limits_i (x_i^{\mathcal{C}}) & \text{If } \min\limits_i(x_i^{\mathcal{C}}) > \min\limits_i(x_i^{H}) - 2\cdot D(H,\mathcal{C}) \\ \min\limits_i (x_i^{\mathcal{C}}) - 2\cdot D(\mathcal{H},\mathcal{C}) & \text{otherwise} \end{cases} \\
y_{TL} &= \begin{cases} \min\limits_i (y_i^{\mathcal{C}}) & \text{If } \min\limits_i(y_i^{\mathcal{C}}) > \min\limits_i(y_i^{H}) - 2\cdot D(H,\mathcal{C}) \\ \min\limits_i (y_i^{\mathcal{H}}) - 2\cdot D(\mathcal{H},\mathcal{C}) & \text{otherwise} \end{cases} \\
x_{BR} &= \begin{cases} \max\limits_i (x_i^{\mathcal{C}}) & \text{If } \max\limits_i(x_i^{\mathcal{C}}) < \max\limits_i(x_i^{H}) + 2\cdot D(H,\mathcal{C}) \\ \max\limits_i (x_i^{\mathcal{H}}) + 2\cdot D(\mathcal{H},\mathcal{C}) & \text{otherwise} \end{cases} \\
y_{BR} &= \begin{cases} \max\limits_i (y_i^{\mathcal{C}}) & \text{If } \max\limits_i(y_i^{\mathcal{C}}) < \max\limits_i(y_i^{H}) + 2\cdot D(H,\mathcal{C}) \\ \max\limits_i (y_i^{\mathcal{H}}) + 2\cdot D(\mathcal{H},\mathcal{C}) & \text{otherwise} \end{cases}
\end{aligned}
\tag{2}
$$

Fig. 7 shows the contours of the component with the respective windows W around the closed cavities.

2.4 Feature vector estimation

The feature vector estimation stage aims at constructing a feature vector corresponding to each cavity pattern, by identifying all segments that belong to a protrusion in the outer contour of each isolated cavity pattern. It is applied in two consecutive modes: a *vertical* and a *horizontal* mode. The vertical mode is used to describe the protrusible segments that may exist either at the top or at the bottom of the character while the horizontal mode is used to describe the protrusible segments that may exist on the right side of the character (see Figs. 8, 9). The feature set which is composed of nine features $F=\{f_1, f_2,..., f_9\}$ expresses the probability of a segment being part of a protrusion. Features $\{f_1, f_2, f_3\}$ describe the protrusible segments that may appear on the top of the character, features $\{f_4, f_5, f_6\}$ describe the protrusible segments that may exist at the bottom of the character and features $\{f_7, f_8, f_9\}$ describe the protrusible segments that may exist on the right side of the character. We have not taken into account segments that may belong to left protrusions, due to our observation that in all cases they correspond to a letter ligament rather than to the main body of a character. A sound example is given in Fig. 7 in the case of Greek character "σ".

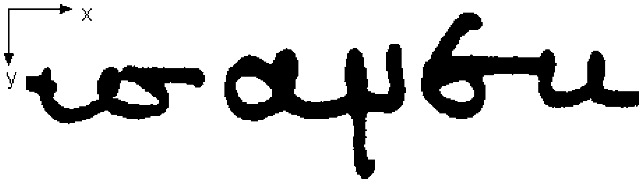


Fig. 6 A part of an image consisting of three connected components each one having one closed cavity

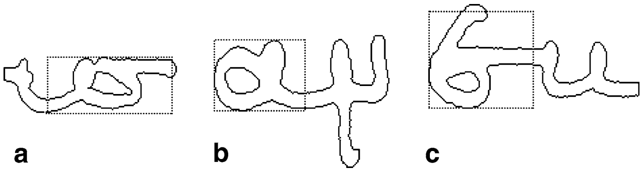


Fig. 7 The contours of the component with the respective adaptive bounding box around each closed cavity. **a** Greek character "σ", **b** Greek character "α", **c** Greek character "ε"

The feature vector is generated in three distinct steps:

2.4.1 Step 1: Bounding box division into blocks

In vertical mode we compute the mean value Y of all $y^{\mathcal{H}}$-coordinates of $\mathcal{H}$ set, which is denoted as:

$$Y = \frac{1}{n}\sum_{i=1}^{n} y_i^{\mathcal{H}} \tag{3}$$

We divide W into three areas of equal width and assign a divide line $F(x)=Y$ as it is shown in Fig. 8a, resulting in six blocks $R_1,...,R_6$.

Furthermore, for the horizontal mode we compute the mean value X of all x^{H}-coordinates of $\mathcal{H}$ set, which is denoted as:

$$X = \frac{1}{n}\sum_{i=1}^{n} x_i^{\mathcal{H}} \tag{4}$$

Similarly, we divide W into three areas of equal height and assign a divide line $F(y)=X$ as it is shown in Fig. 9a, resulting in extra six blocks $\{R_7,...,R_{12}\}$.

2.4.2 Step 2: Block correction

In order to describe the pixels of a protrusible segment, ignoring the other pixels of the character, the process must be restricted to the upper side of blocks R_{i} with $i\in[1,3]$, to the lower side of blocks R_{i} with $i\in[4,6]$, and to the right most side of blocks R_{i} with $i\in[7,9]$. Blocks R_{10}, R_{11}, R_{12} are not used for feature extraction because any protrusion in these regions is a letter ligature rather than a dominant part of the character. Thus, for each block, we compute an offset (P_{i}) to the divide line at the corresponding mode (see Figs. 8b, 9b). This leads to a new assignment of the block area that is different for each block that initially partitioned the bounding box. The

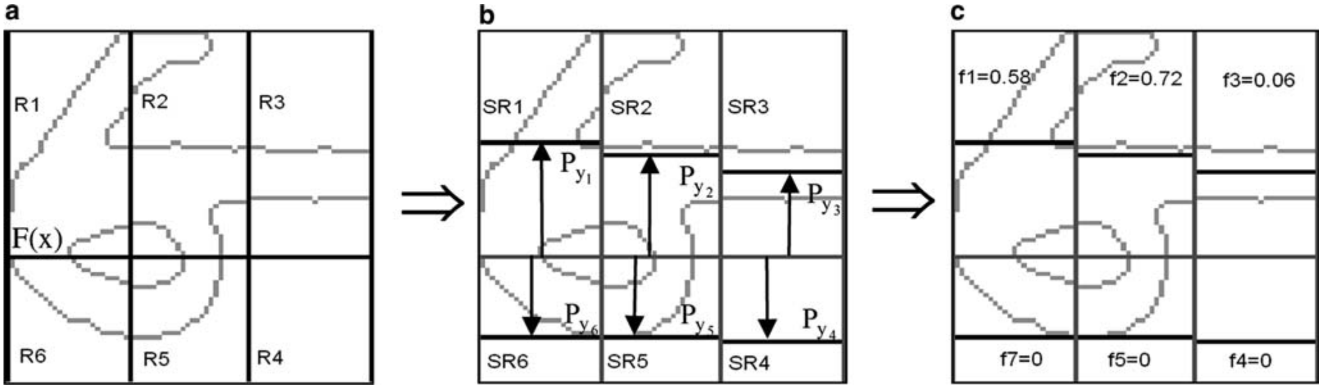


Fig. 8 Vertical mode. **a** Region R_1, R_2,...,R_6, defined by the $F(x)$. **b** Sub-regions SR_1, SR_2,...,SR_6 defined by $P_{\mathrm{y}i}$ offsets respectively. **c** The values of features f_1...f_6 for the segmented Greek letter "ε"

corrected blocks will delimit the area for the block-based feature computation.

The offset, at the vertical mode $\left(P_{y_i}\right)$ and the horizontal mode (P_{x_i}) is calculated as follows:

$$P_{y_i} = \begin{cases} \min\limits_j (y_j^{\mathcal{H}_{R_i}}) - D(\mathcal{H}_{R_i}, \mathcal{C}) & \text{if } (\mathcal{H}_{R_i} \neq \text{null}) \vee (i \in [1,3]) \\ \max\limits_j (y_j^{\mathcal{H}_{R_i}}) + D(\mathcal{H}_{R_i}, \mathcal{C}) & \text{if } (\mathcal{H}_{R_i} \neq \text{null}) \vee (i \in [4,6]) \\ \mathcal{H}_{R_i} & \text{if } \mathcal{H}_{R_i} = \text{null} \end{cases}$$
$$P_{x_i} = \begin{cases} \max\limits_j (x_j^{\mathcal{H}_{R_i}}) + D(\mathcal{H}_{R_i}, \mathcal{C}) & \text{if } (\mathcal{H}_{R_i} \neq \text{null}) \ \vee (i \in [7,\ 9]) \\ D(\mathcal{H}_{R_i}, \mathcal{C}) & \text{if } \mathcal{H}_{R_i} = \text{null} \end{cases} \tag{5}$$

where $\mathcal{H}_{R_i} = \{\ (x_j^{\mathcal{H}_{R_i}}, y_j^{\mathcal{H}_{R_i}}) \subset\ \mathcal{H}, j \in [1, n_{R_i}], i \in [1, 9]\}$ are the sets of the pixel coordinates, which constitute the part of the closed cavity contour depicted in block R_{i}.

2.4.3 Step 3: Block-based feature computation

In this step, we estimate the directions of the contour pixels by taking into account pairs of adjacent pixels during a clockwise tracing of the outer contour. Hence, for each pixel j of the contour we determine the local orientation of the contour S_j, taking nominal values from the set $\{W, SW, S, SE, E, NE, N, NW\}$. Once the directions are evaluated the proposed feature f_{i} is defined as follows:

$$f_i = \frac{1}{D(\mathcal{H}, \mathcal{C})} \sum_{j=1}^{m_i} g_i\left(s_j^i\right) \tag{6}$$

where $g_{\mathrm{i}}(\cdot)$ is a function depending on the orientation of the pixel and the block considered and m_{i} is the total number of pixels of the outer contour inside block SR_{i}. The term $D(\mathcal{H}, \mathcal{C})$ is used as a normalization factor allowing the feature to be invariant with respect to character scaling. The $g_{\mathrm{i}}(\cdot)$ values are explicitly defined in

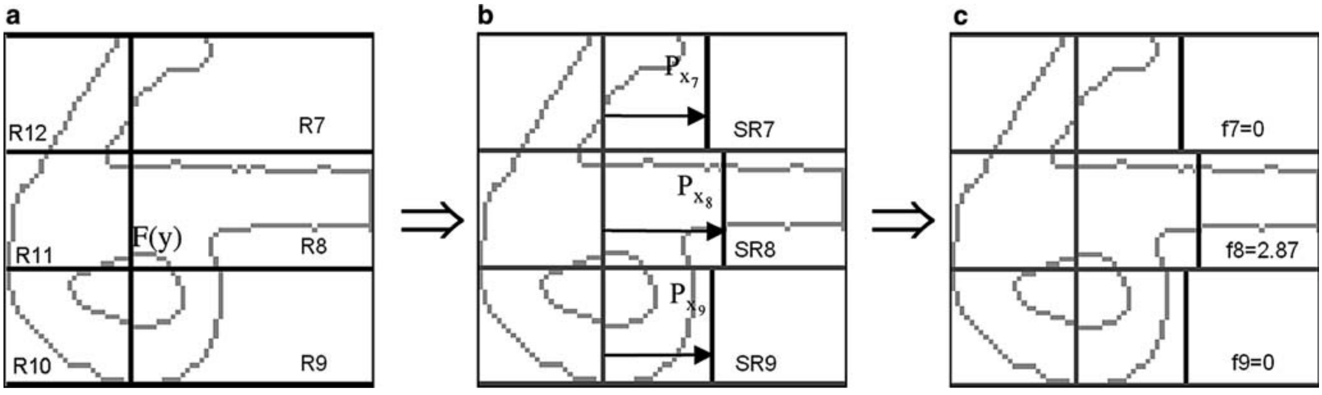


Fig. 9 Horizontal mode. **a** Regions R_7, R_8, R_9 defined by $F(y)$. **b** Sub-regions SR_7, SR_8, SR_9 defined by the $P_{\mathrm{x}i}$ offsets respectively. **c** The values of features f_7, f_8, f_9 for the segmented Greek letter "ε"

Table 2 $g_i(\cdot)$ in 8-connectivity contour following

	E	NE	N	NW	W	SW	S	SE
g_1	0	0	1	1	0	1	1	1
g_2	0	1	1	1	0	1	1	1
g_3	0	1	1	1	0	1	1	0
g_4	0	1	1	1	0	0	1	1
g_5	0	1	1	1	0	1	1	1
g_6	0	1	1	0	0	1	1	1
g_7	1	1	0	1	1	1	0	0
g_8	1	1	0	1	1	1	0	1
g_9	1	1	0	1	1	0	0	1

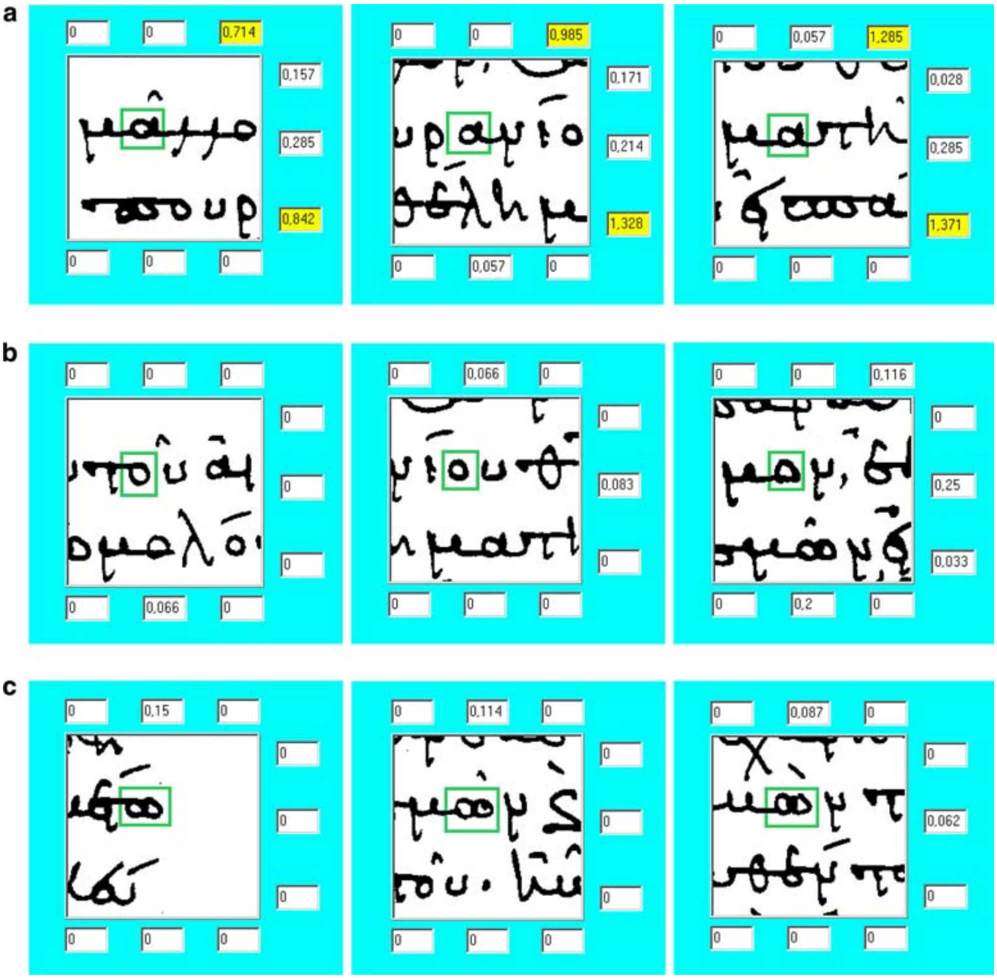
↖	N	↗
W	•	E
↙	S	↘

Table 2. They determine a unique pixel template for each block *i* that expresses the expected local orientation of the corresponding contour pixels. During contour following, $g_i(\cdot)$ equals 1 when the examined pixel belongs to either a vertical or a horizontal protrusible segment, otherwise it equals 0. Extension of the above procedure for characters having more than one closed cavity is straightforward. Multiple closed cavities are treated as one after a merging process which preserves the initially topology.

An example of the estimated features for characters having one or two closed cavities is given in Fig. 10. In this figure, our graphical user interface that concern feature estimation, is shown. One may observe a set of nine cells which enable the visualization of feature values. Each cell position corresponds to the position of the respective protrusible segment in the set *F*. A pseudo-code of the above procedure is given in Fig. 11.

3 Experimental results

In this section, we describe the experimental procedure used to evaluate the proposed method, when applied to old Greek manuscripts, and discuss the experimental results. The corpus used for the experiments as well as its training and testing set partitions are analyzed in Sect. 3.1. Due to the modularity of the proposed algorithm, two distinct kinds of experiment sets were conducted: one for closed-cavity pattern detection and a second for cavity-based character recognition, based on previously found closed-cavity patterns. These experiments are presented separately in Sect. 3.2 and 3.3, respectively.



a:
0 0 0,714 | 0,157 0,285 0,842 | 0 0 0
0 0 0,985 | 0,171 0,214 1,328 | 0 0,057 0
0 0,057 1,285 | 0,028 0,285 1,371 | 0 0 0

b:
0 0 0 | 0 0 0 | 0 0,066 0
0 0,066 0 | 0 0,083 0 | 0 0 0
0 0 0,116 | 0 0,25 0,033 | 0 0,2 0

c:
0 0,15 0 | 0 0 0 | 0 0 0
0 0,114 0 | 0 0 0 | 0 0 0
0 0,087 0 | 0 0,062 0 | 0 0 0

Fig. 10 Feature estimation examples: **a** features for characters "α", **b** features for characters "o" and (c) features for characters "ω"

Fig. 11 Pseudocode of the feature vector generation algorithm

```
Feature Extraction Pseudo-code
//Bounding box division into blocks
• Set $Y = \frac{1}{n}\sum_{i=1}^{n} y_i^{\mathcal{H}}$
• Divide $W$ into three areas of equal width and assign a divide line $F(x)=Y$ resulting in 6 blocks $R_1,\dots,R_6$.
• Set $X = \frac{1}{n}\sum_{i=1}^{n} x_i^{\mathcal{H}}$
• Divide W into three areas of equal height and assign a divide line F(y)=X resulting in extra 6 blocks {R7,…,R12}.

• //Block Correction
• For i=1 to 7
    • If $\mathcal{H}_{R_i}$ = null then
        • Set $P_{y_i} = D(\mathcal{H},C)$
    • Else
        • If i<=3 then
            • Set $P_{y_i} = \min_j(y_j^{\mathcal{H}_{R_i}}) - D(\mathcal{H}_{R_i},C)$
        • Else
            • Set $P_{y_i} = \max_j(y_j^{\mathcal{H}_{R_i}}) + D(\mathcal{H}_{R_i},C)$
        • Endif
    • Endif
• Next i
• For i =7 to 9
    • If $\mathcal{H}_{R_i}$ = null then
        • Set $P_{x_i} = D(\mathcal{H},C)$
    • Else
        • Set $P_{x_i} = \max_{\mathrm{j}}(x_j^{\mathcal{H}_{R_i}}) + D(\mathcal{H}_{R_i},C)$
    • Endif
• Next i

• // Feature computation
• For each block $S_{R_i}$
    • Set $f_i = 0$
    • For each pixel j of the outer contour inside block $S_{R_i}$
        • Set $f_i = f_i + \frac{1}{D(\mathcal{H},C)} g_i(s_j^i)$
    • Next j
• Next i
• End of Algorithm
```

3.1 The corpus

The corpus used for the experiments originates from three different writers of the Book of Job collection (see Sect. 1), henceforth called wr1, wr2 and wr3. The inclusion of different writers allowed testing the robustness of the proposed method against different writing styles. To reliably assess our method, we manually annotated the corpus, by labeling all instances of characters—or character ligatures—containing closed

Table 3 The dictionary of closed cavity patterns including the number of pattern occurrences

Pattern ID	1	2	3	4	5	6
Pattern	o	o o	o o o	o o o o	o o	o o o
Occurrences	786	130	7	3	30	11

Table 4 Training set/Test set configuration for the CL1 and CL2 datasets

ID	Samples	Training	Test
CL1-1	754	70 (10% wr1, wr2)	684 (90% wr1, wr2)
CL1-2	754	147 (20% wr1, wr2)	607 (80% wr1, wr2)
CL1-3	479	147 (20% wr1, wr2)	332 (100% wr3)
CL1-4	402	70 (10% wr1, wr2)	332 (100% wr3)
CL1-5	1086	754 (100% wr1, wr2)	332 (100% wr3)
CL2-1	123	12 (10% wr1, wr2)	111 (90% wr1, wr2)
CL2-2	123	24 (20% wr1, wr2)	99 (80% wr1, wr2)
CL2-3	73	12 (10% wr1, wr2)	61 (100% wr3)
CL2-4	85	24 (20% wr1, wr2)	61 (100% wr3)
CL2-5	184	123 (100% wr1, wr2)	61 (100% wr3)

Table 5 Recall/Precision for the characters or character ligatures in each of the closed cavity patterns

ID	Recall (%)	Precision (%)
1	95.81	97.42
2	94.61	86.62
3	100.00	53.85
4	100.00	100.00
5	84.37	96.43
6	87.50	100.00

cavities, arranged in some pattern, as discussed in Sect. 2. Table 3 shows the number of occurrences of each cavity pattern in the dataset. Notice that the vast majority of patterns are those having either one or two adjacent closed cavities. Overall, the number of characters and character ligatures labeled were 967 and they approximately correspond to 60% of the complete character set.

Since our ultimate goal is applying the proposed algorithm to an important number of documents, the experiments were oriented towards assessing how robust the method is against unseen data and how much training is required. Since the cavity pattern detection method does not require any training, the last issue concerns primarily the assessment of the proposed feature generation method. Namely, we assessed our method using five different partitioning scenarios, summarized in Table 4. The first two scenarios consisted in training using 10% (respectively 20%) of the characters of the first two authors and testing using the other 90% (respectively 80%) of the same authors. The other three, consisted in training using a part of the data coming from the first two authors (10, 20 and 100%) and testing using the totality of data coming from the third author.

3.2 Evaluation of the closed cavity pattern detection method

The first experiment was conducted to test the performance of the closed cavity pattern detection algorithm. Since this algorithm does not require any training, there had been no need for using the abovementioned scenarios. Instead, all the available labeled samples were used as a test set. We recall that the dataset involved in our experiments have been preprocessed with a binarization and image enhancement algorithm, as it described

Table 6 Algorithmic performance for the CL1 dataset. Numbers in parenthesis represent the parameters used for achieving the optimal scores. The ID column corresponds to the different scenarios as shown in Table 4. For the SVM kernel, the number of support vectors found is also given

ID	KNN-L1	KNN-L2	SVM-rbf
CL1-1	90.49 ($k=1$)	90.78 ($k=1$)	93.42 ($\gamma=0.94$, $c=50$, $SVs=45$)
CL1-2	94.06 ($k=1$)	93.73 ($k=1$)	95.05 ($\gamma=0.98$, $c=50$, $SVs=62$)
CL1-3	97.89 ($k=2$)	97.59 ($k=2$)	97.89 ($\gamma=0.04$, $c=50$, $SVs=63$)
CL1-4	94.27 ($k=1$)	96.08 ($k=1$)	97.28 ($\gamma=0.2$, $c=100$, $SVs=42$)
CL1-5	98.19 ($k=10$)	98.19 ($k=4$)	98.49 ($\gamma=0.8$, $c=1$, $SVs=216$)

Table 7 Algorithmic performance for the CL2 dataset. Numbers in parenthesis represent the parameters used for achieving the optimal scores. The ID column corresponds to the different scenarios as shown in Table 4.

ID	KNN-L1	KNN-L2	SVM-rbf
CL2-1	95.49 ($k=1$)	93.69 ($k=1$)	94.59 ($\gamma=0.1$, $c=10$, $SVs=9$)
CL2-2	93.93 ($k=1$)	92.92 ($k=1$)	94.94 ($\gamma=0.1$, $c=20$, $SVs=10$)
CL2-3	100.0 ($k=1$)	100.0 ($k=1$)	100.0 ($\gamma=0.1$, $c=10$, $SVs=9$)
CL2-4	98.36 ($k=6$)	95.99 ($k=1$)	100.0 ($\gamma=0.1$, $c=10$, $SVs=11$)
CL2-5	96.72 ($k=1$)	98.36 ($k=1$)	100.0 ($\gamma=0.1$, $c=10$, $SVs=24$)

Table 8 Confusion matrix for the scenario CL1-1.

	α	ε	ο	σ	ρ	δ
α	170	9	0	0	1	1
ε	6	119	0	0	0	1
ο	1	0	155	0	2	0
σ	5	6	0	106	0	0
ρ	0	0	3	0	65	0
δ	0	2	9	0	0	22

Table 9 Confusion matrix for the scenario CL2-1.

	π	εσ	ω
π	48	0	6
εσ	0	2	0
ω	0	0	55

in Sect. 2.1. Thus, detection was conducted on images of improved quality (see Fig. 3).

Table 5 shows the results obtained by applying the algorithm, indicating the recall and the precision rates for each one of the closed cavity patterns. Recall is the number of correct closed-cavity patterns found divided by the total number of existing closed-cavity patterns. Precision is the number of correct closed-cavity patterns found divided by the total number of closed-cavity patterns found. The overall accuracy on cavity pattern detection is 95.2%.

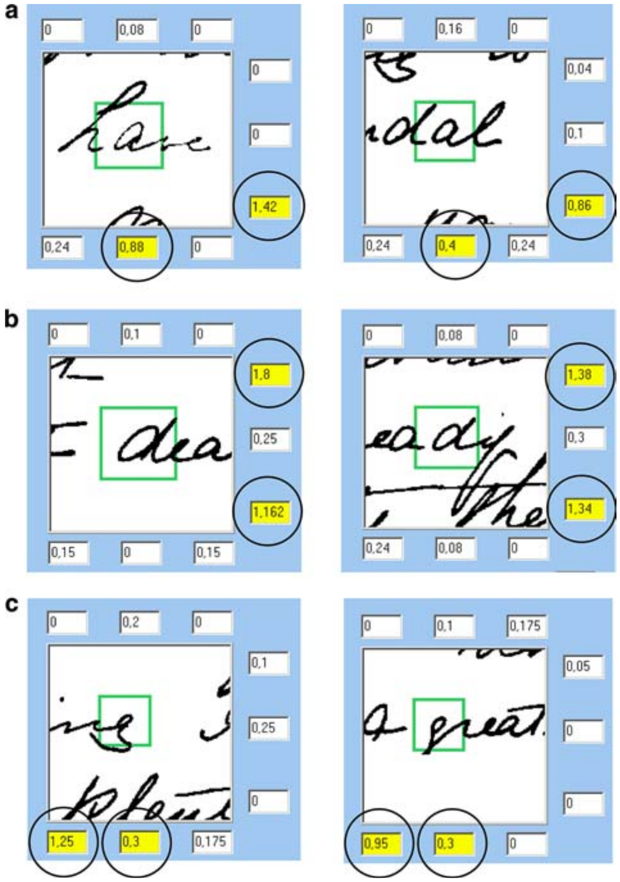
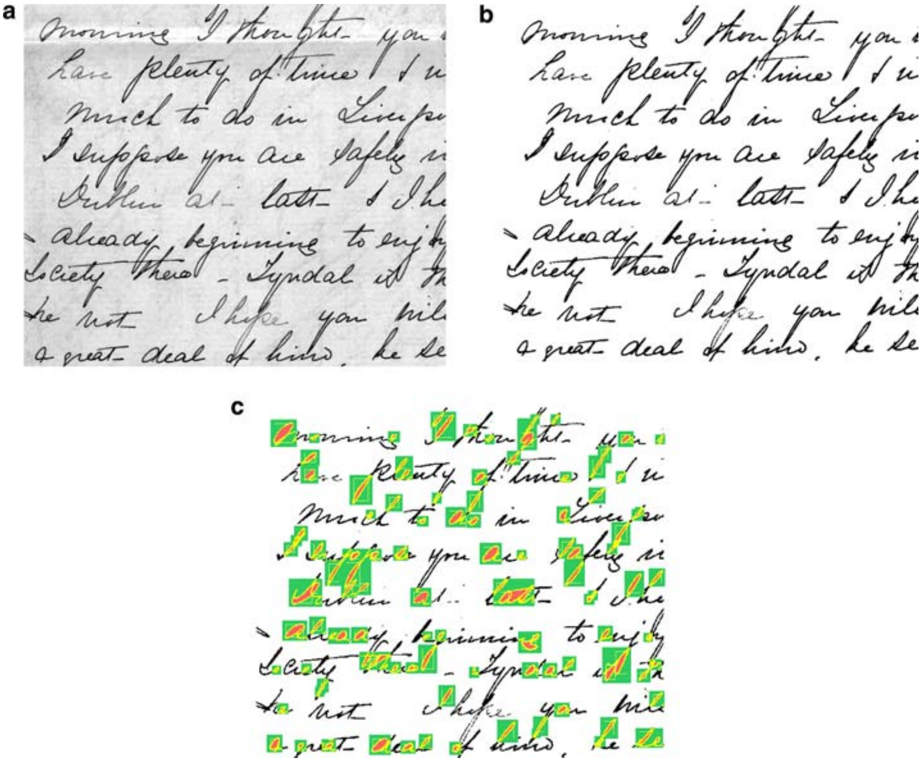
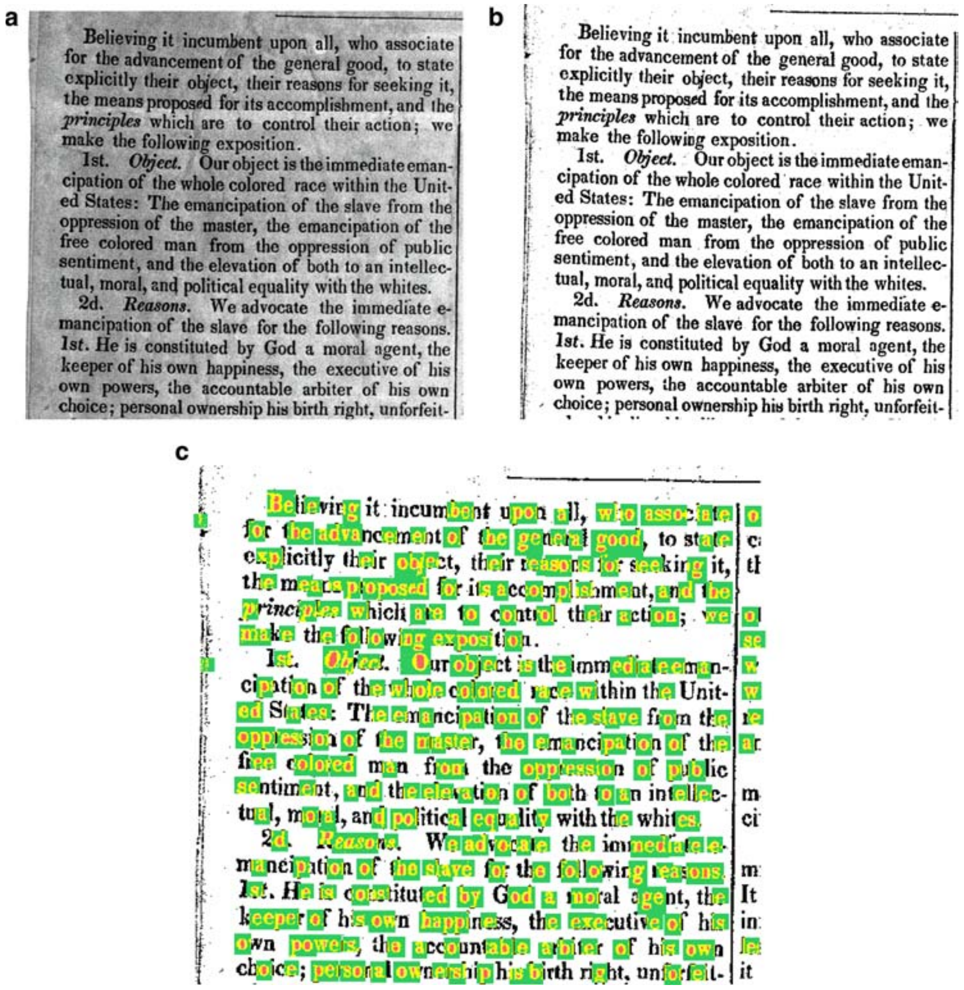


Fig. 13 Feature estimation: **a** features for characters “a”, **b** features for characters “d”, **c** features for characters “g”

Fig. 12 Example of Application (I): **a** original gray scale image, **b** B/W image after enhancement and **c** estimated areas that contain closed cavities





Believing it incumbent upon all, who associate for the advancement of the general good, to state explicitly their object, their reasons for seeking it, the means proposed for its accomplishment, and the *principles* which are to control their action; we make the following exposition.

1st. *Object.* Our object is the immediate emancipation of the whole colored race within the United States: The emancipation of the slave from the oppression of the master, the emancipation of the free colored man from the oppression of public sentiment, and the elevation of both to an intellectual, moral, and political equality with the whites.

2d. *Reasons.* We advocate the immediate emancipation of the slave for the following reasons. 1st. He is constituted by God a moral agent, the keeper of his own happiness, the executive of his own powers, the accountable arbiter of his own choice; personal ownership his birth right, unforfeit-

Believing it incumbent upon all, who associate for the advancement of the general good, to state explicitly their object, their reasons for seeking it, the means proposed for its accomplishment, and the *principles* which are to control their action; we make the following exposition.

1st. *Object.* Our object is the immediate emancipation of the whole colored race within the United States: The emancipation of the slave from the oppression of the master, the emancipation of the free colored man from the oppression of public sentiment, and the elevation of both to an intellectual, moral, and political equality with the whites.

2d. *Reasons.* We advocate the immediate emancipation of the slave for the following reasons. 1st. He is constituted by God a moral agent, the keeper of his own happiness, the executive of his own powers, the accountable arbiter of his own choice; personal ownership his birth right, unforfeit-

Fig. 14 Example of Application (II): **a** original gray scale image, **b** B/W image after enhancement and **c** estimated areas that contain closed cavities

3.3 Evaluation of character and character ligature recognition

The second set of experiments was conducted to evaluate the feature vector generation method. The feature vector is used to classify a previously detected closed-cavity pattern to a character or character ligature. Evaluation was done by measuring the classification accuracy of two popular classification algorithms trained and tested using the generated feature vectors: the K-NN [32] classifier and support vector machines (SVMs) using the rbf kernel [33, 34]. K-NN was used in two variants, with L1 norm and L2 norm. The use of several essentially nonparametric classifiers was made to ensure that evaluation of the feature vector does not depend on a particular classification scheme. Moreover, a systematic search took place in order to determine the optimum values of parameters, i.e., the number of neighbors (k) for the K-NN algorithm and the rbf-variance (γ) and cost (c) parameter of the SVM.

Notice that the feature vector step is important only when several characters correspond to the same cavity pattern. Namely, the last three cavity patterns listed in Table 1 correspond one-by-one to some specific character or character ligature and, hence, there is no need for further classification. As a result, the experiments concentrated on the classification of the first two cavity patterns listed in Table 1 (IDs 1 and 2, respectively), which do correspond to a variety of characters. Namely, two distinct classification tasks were defined: (a) classify pattern with ID 1 to one among six characters (α, o, ε, σ, ρ, δ) and (b) classify patterns with ID 2 to one among 3 characters (π, ω, $\varepsilon\sigma$). Each task was tested using five different training/testing partitions, as discussed in Sect. 3.1.

The summary of classification results for both the tasks using each partitioning scenario along with the optimum classifier parameters, are listed in Tables 6 and 7, respectively. Notice that the scores achieved in both datasets were very high even in cases where the samples were few. This particular aspect is very encouraging, since it proves the good generalization performance of the algorithms.

To gain further insight, we present two confusion matrices corresponding to the first and second classification tasks, resulting by testing the SVM classifier using the first partitioning scenario. The matrices, shown in Tables 8 and 9, indicate the points of confusion between

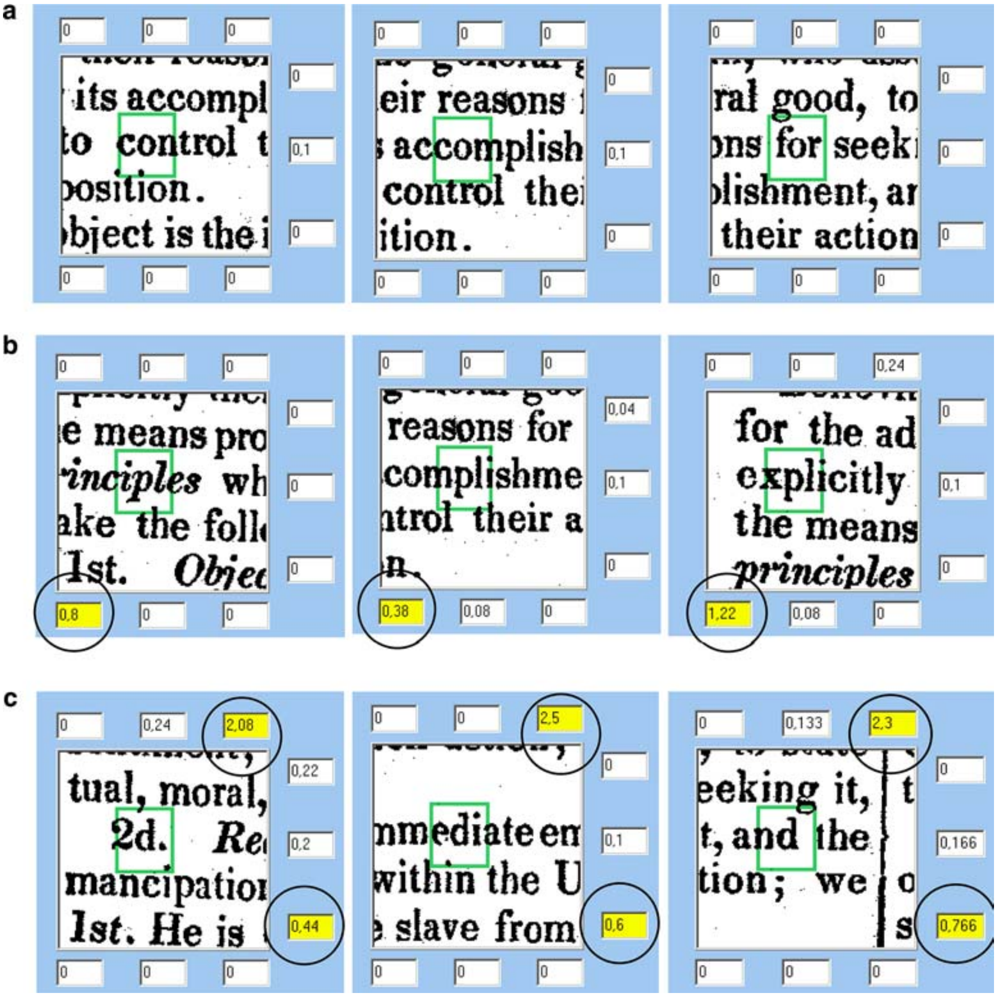


Fig. 15 Feature estimation: **a** features for characters “o”, **b** features for characters “p”, **c** features for characters “d”

characters. In particular, Table 8 shows the confusion matrix for the first classification task. It can be noticed that we get certain cases of misclassifications. In particular, characters “α” and “ε”, are mutually misclassified nine (9) and six (6) times, respectively. Character “δ” is misclassified as “o” nine (9) times, while character “σ” is misclassified as “α” and “ε” five (5) and six (6) times, respectively. Table 9 shows the confusion matrix for the second classification task. It can be noticed that the only misclassification occurs for character “π” which is classified as “ω” six (6) times.

4 Other applications

The proposed methodology can be applied to several OCR tasks where conventional recognition techniques may fail. These tasks mainly concern document images that cannot be easily segmented into words or letters. In this section, we present three representative examples of applying our segmentation-free methodology in order to assist recognition. More specifically, we examine the following applications:

(I) Cursive handwritten document recognition;
(II) Low-quality machine printed document recognition;
(III) Reading text information from maps.

In Fig. 12, a representative cursive handwritten document for Application (I) is presented. After binarization and image enhancement (Fig. 12b), all the closed cavities are marked (Fig. 12c). As we can observe from Fig. 13, several letters such as “a”, “d” and “g” can be successfully recognized since the dominant estimated features (depicted in circles in Fig. 13) can provide the desired discrimination.

As a representative example for Application (II), a low-quality machine printed document was selected from the old newspaper collection of the Library of Congress [35]. As we can see in Fig. 14, after binarization and image enhancement, we get a B/W image that contains several closed cavities. Suitability of the extracted features is exemplified in Fig. 15.

Application (III) concerns text extraction and recognition from maps. The proposed algorithm has been proved very efficient for this application since the

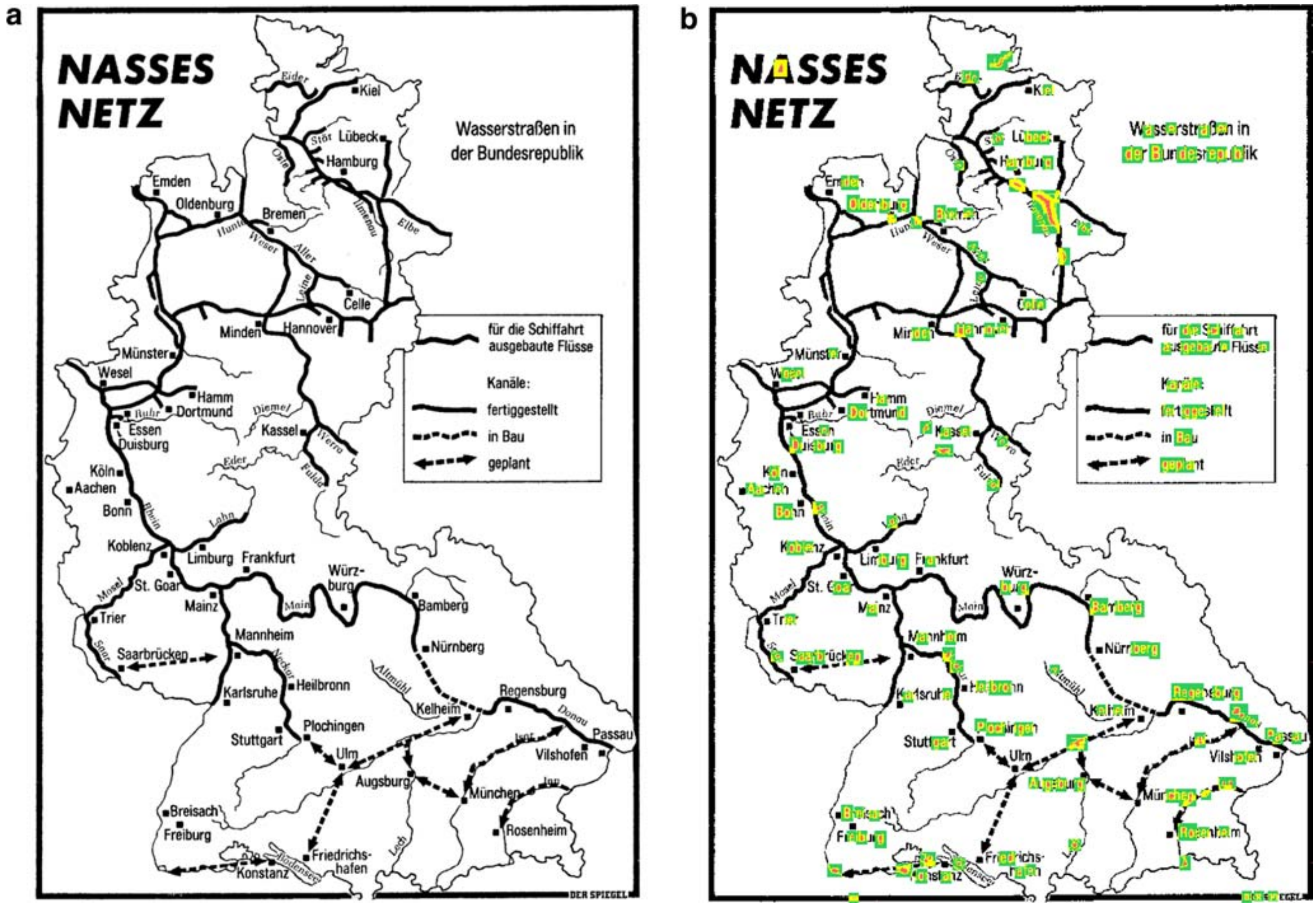


Fig. 16 Example of Application (III): **a** original image, **b** estimated areas that contain closed cavities

segmentation and text location task is very difficult to be applied. In Fig. 16, the closed cavities detection algorithm is demonstrated on a map image coming from the collection of MediaTeam Document Database II [36]. Suitability of the extracted features is exemplified in Fig. 17.

5 Conclusions and future work

In this paper, we present a novel methodology that assists recognition of early Christian Greek manuscripts. We do not provide a solution for a complete character recognition system but we strive toward an assessment of the recognition procedure by tracing and recognizing the most frequently appearing characters or character ligatures, using a segmentation-free, quick and efficient approach. Based on the observation that closed cavities appear in the majority of characters and character ligatures, we propose a recognition technique that consists of several distinct stages. Experimental results show that the proposed method gives highly accurate results that offer a great assistance to old Greek handwritten manuscript interpretation. Additionally, applying our methodology to other OCR applications such as cursive handwritten recognition, low-quality machine printed character recognition or text location and recognition in map images, we not only prove the robustness of the proposed method but we also demonstrate its generic flavor in case that segmentation and text location tasks are very difficult to perform.

Future work involves the detection and recognition of the remaining old Greek handwritten character and character ligatures that do not include closed cavities, as well as the testing of the performance of the proposed technique for other types of old handwritten historical manuscripts.

Acknowledgements This research is carried out within the framework of the Greek GSRT-funded R&D project, D-SCRIBE, which aims to develop an integrated system for digitization and processing of old Greek manuscripts.

References

1. Vinciarelli A (2002) survey on off-line Cursive Word Recognition. Pattern Recognition 35:1433–1446
2. Lu Y, Tan CL (2002) Combination of multiple classifiers using probabilistic dictionary and its application to postcode recognition. Pattern Recognition 35:2823–2832

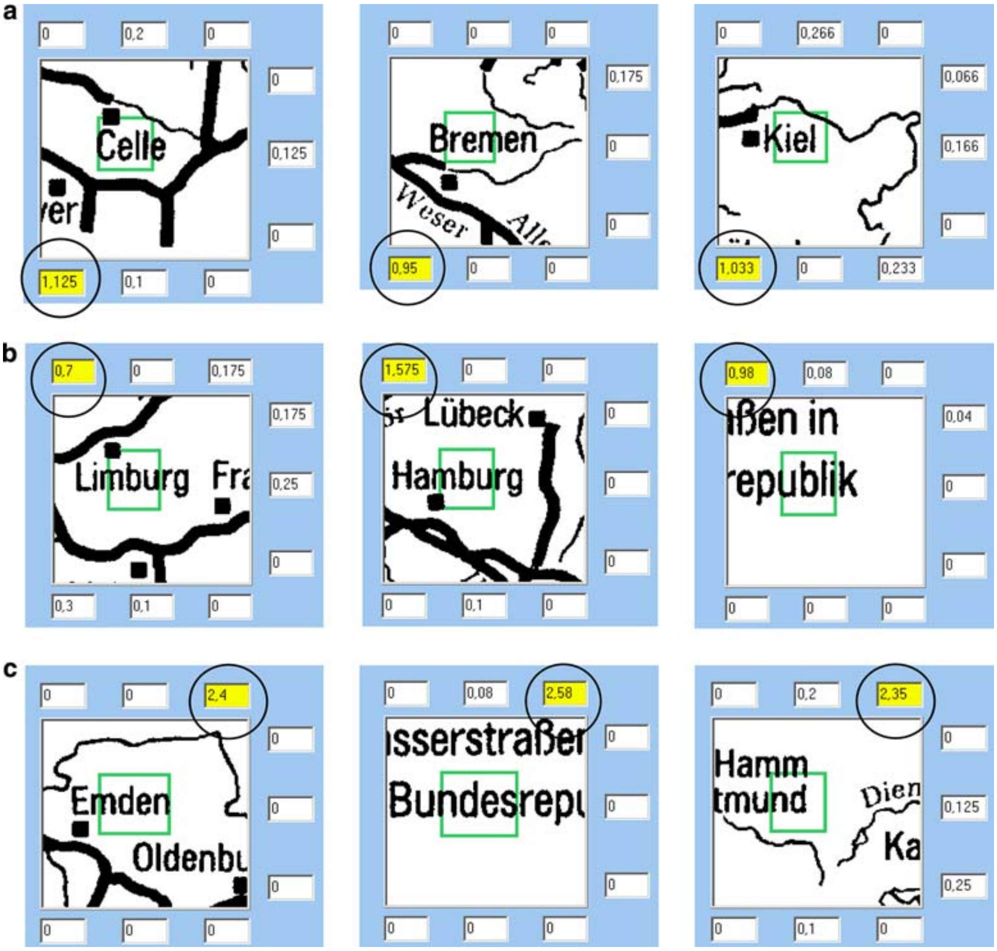


Fig. 17 Feature estimation: **a** features for characters "e", **b** features for characters "b", **c** features for characters "d"

3. Brakensiek A, Rottland J, Rigoll G (2003) Confidence measures for an address reading system. Seventh international conference on document analysis and recognition, ICDAR2003, pp 294–298
4. Hirano T, Okada Y, Yoda F (2001) Field extraction method from existing forms transmitted by facsimile. Sixth international conference on document analysis and recognition, ICDAR2001, pp 738–742
5. Xu Q, Lam L, Suen CY (2001) A knowledge-based segmentation system for handwritten dates on bank cheques. Sixth international conference on document analysis and recognition, ICDAR2001, pp 384–388
6. Gorski N, Anisimov V, Augustin E, Baret O, Price D, Simon JC (1999) A2iA check reader: a family of bank check recognition systems. Proc. fifth int'l conf. document analysis and recognition, pp 523–526
7. Suen CY, et al (1993) Building a new generation of handwriting recognition systems. Patt Recog Lett 14:303–315
8. Guillevic D, Suen CY (1997) HMM word recognition engine. Fourth international conference on document analysis and recognition ICDAR97, pp 544
9. Kavallieratou E, Fakotakis N, Kokkinakis G (2002) Handwritten character recognition based on structural characteristics. 16th International conference on pattern recognition, pp 139–142
10. Eastwood B et al. (1997) A feature based neural network segmenter for handwritten words. International conference on computational intelligence and multimedia applications (ICCIMA'97), Australia, pp 286–290
11. Lu Y, Shridhar M (1996) Character segmentation in handwritten words—an overview, Patt Recog 29(1):77–96
12. Xiao X, Leedham G (1999) Cursive script segmentation incorporating knowledge of writing. Proceedings of the fifth international conference on document analysis and recognition, pp 535–538
13. Plamondon P, Privitera CM (1999) The segmentation of cursive handwritten: an approach based on off-line recovery of the motor-temporal information, IEEE Trans Image Process 8:80–91
14. Chi Z, Suters M, Yan H (1995) Separation of single-and double-touching handwritten numeral strings. Opt Eng 34:1159–1165
15. Zhao S, Chi Z, Shi P, Yan H (2003) Two-stage segmentation of unconstrained handwritten Chinese characters. Pattern Recognition 36:145–156
16. Farag R (1979) Word-level recognition of cursive script, IEEE Trans. Comput Vol C-28:172–175
17. Simon J (1992) Off-line cursive word recognition. Proceedings of the IEEE 80:1150–1161
18. Madhvanath S, Govindaraju V (1993) Holistic lexicon reduction. Proceedings of the Third International Workshop on Frontiers in Handwriting Recognition. Buffalo, N.Y:71–82
19. Madhvanath S, Kleinger E, Govindaraju V (1999) Holistic verifications of handwritten phrases. IEEE Trans. PAMI 21:1344–1356
20. Chen CH, de Curtins J (2003) Word Recognition in a Segmentation-Free Approach to OCR. Second International Conference on Document Analysis and Recognition (ICDAR'93), pp 573–576
21. Chen CH, de Curtins J (1992) A Segmentation-free Approach to OCR. IEEE Workshop on Applications of Computer Vision, pp 190–196

22. Duda R, Hart E (1973) Pattern Classification and Scene Analysis. Wiley
23. Amin A and Masini G Machine recognition of cursive Arabic words, Application of Digital Image Processing IV, San Diego, CA, August 1982, Vol SPIE-359, pp.286–292]
24. Mori S, Suen CY, Yamamoto K Historical review of OCR research and development, Proc. IEEE, vol. 80 1992, pp. 1029–1058
25. Ulmann J. R. Experiments with the n-tuple method of pattern recognition, IEEE Trans. Computers, vol 18, no 12,1969 pp. 1135–1137
26. Jung DM, Krishnamoorty MS, Nagy G, Shapira A. N-tuple features for OCR revisited, IEEE Trans. PAMI vol. 18, no. 7,1996, pp. 734–745
27. Gonzalez RC, Woods RE (1992) Digital Image Processing. Addison-Wesley
28. Gatos B, Pratikakis I, Perantonis SJ Locating Text in Historical Collection Manuscripts. Lecture Notes on AI, SETN 2004, pp. 476–485
29. Niblack W (1986) An Introduction to Digital Image Processing. Prentice Hall, Englewood Cliffs NJ, pp 115–116
30. Pavlidis T (1992) Algorithms for Graphics and Image Processing. Computer Science Press, Rockville, MD
31. Xia F (2003) Normal vector and winding number in 2D digital images with their application for hole detection. Pattern Recognition 36:1383–1395
32. Jain A (1989) Fundamentals of digital image processing. Prentice Hall
33. Theodoridis S, Koutroumbas K (1997) Pattern Recognition. Academic Press
34. Chang CC, Lin, C. J. LIBSVM: A library for support vector machines 2001, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm
35. American Memory: Historical Collections for the National Digital Library, http://memory.loc.gov/
36. Sauvola J, Kauniskangas H (1999) MediaTeam Document Database II, a CD-ROM collection of document images. University of Oulu, Finland