Contributed Paper

Identification of Text-Only Areas in Mixed-Type Documents

C. STROUTHOPOULOS
Democritus University of Thrace, Greece

N. PAPAMARKOS
Democritus University of Thrace, Greece

C. CHAMZAS
Democritus University of Thrace, Greece

(Received July 1996; in revised form December 1996)

The identification of text areas in a document is crucial for optical character recognition (OCR), image-compression and image-storage systems. This paper presents a new method for text identification in mixed-type documents. This type of document contains text, drawings and halftones. The proposed method separates the document into text and non-text regions. Thus, the objective is to find, with confidence, the text region of the documents. The method is based on text characteristics such as size, frequency, collinearity and vicinity of connected components, while in the final stage a new texture-analysis technique is applied. For collinearity and vicinity checking, a new technique is used, that overcomes the difficulties of the application of Hough transform. The proposed segmentation method belongs to the bottom-up categories, and is more robust than other techniques. It can identify text regions in difficult cases such as skewed documents, non-rectangular text regions, or text included in drawings or halftone regions. The performance of the method was tested on a variety of images. Its effectiveness is demonstrated by several typical examples.

Keywords: Document image analysis, segmentation, page layout analysis, OCR.

1. INTRODUCTION

Most digitized documents contain halftone pictures and line drawings, along with text. An important procedure in the digital processing of documents is page layout analysis. The goal of the page layout analysis is to discover the formatting of the text and, from that, to derive the meaning associated with the positional and functional blocks in which the text is located (O’Gorman and Kasturi, 1995; Fujisawa et al., 1992; Schurmann et al., 1992). As a result, it labels the parts of a document as text, halftones or line drawings (Pavlidis and Zhou, 1992). Page layout analysis is an important preprocessing procedure for many other applications. For example, in optical character recognition (OCR) the segmentation is a necessary pre-processing procedure. Also, the compression ratio can be improved if the text and image areas are encoded using different methods. The best archiving of mixed-type documents using block segmentation and recognition also requires text-area definition (Chauvet, 1993). Additionally, the identification of text areas is also important for other special applications such as CAD/CAM and communication systems.

There are many problems in document page-layout analysis: identification of image areas, separation of characters included in an image, identification and extraction of text-blocks, and separation of overlapping and touching characters. This paper deals with the problem of the automatic identification of text areas of a document, and the proposed segmentation method answers the question “Where in the document do we have only text?”

In the literature there are three basic segmentation approaches: top-down (or model-driven), bottom-up (or
data-driven), and hybrid (Pavlidis and Zhou, 1992). In top-down methods, a document is segmented from large components (high-level) to smaller, more detailed, subcomponents (lower-level). Most of top-down techniques are based on the run length smoothing (RLS) algorithm (Wong et al., 1982), also called the constrained run length (CRL) algorithm (Wahl et al., 1989), and the projection profile cuts (Wang and Shihari, 1989). The RLS method imposes a smoothing on the binary form of the document using two predetermined parameters (one for the vertical and one for the horizontal direction) defining the document blocks. By using this method, it is possible to segment even handwritten text blocks (Witten et al., 1994). For the block classification, additional parameters (defined in a heuristic way) are used, leading to the necessity to train the system with documents having similar fonts or other common morphological characteristics. A primary advantage of the RLS method is that it can achieve layout analysis fast. For most page formats, this is a very effective approach. It must be noted that the method is not robust since, if the assumptions made to determine the heuristic parameters are not satisfied, the method will fail. Another disadvantage of this method is that the page must be separable into blocks by horizontal and vertical cuts [Manhattan layout (O’Gorman and Kasturi, 1995)]. Hence, for pages where text does not have linear bounds, and where graphics are intermixed both in and around text, these methods will fail.

Bottom-up methods involve the grouping of pixels as connected components (marks) and merge these components into successively larger regions. In one of them, Fletcher and Kasturi (1988), Kasturi et al. (1990), and Kasturi and Trivedi (1990) proposed a procedure that starts by first finding the connected components of an image, and then separating the graphics from the text using the relative frequency of the occurrence of components as a function of their areas. In the next step, they use an iterative procedure to improve the initial estimation by applying the Hough transform to all connected components. The method performs well if the initial document conforms to certain requirements about the size of characters, interline spacing, character spacing, and resolution. This method is quite complex and, as the authors have reported, computationally expensive. Another bottom-up technique proposed by O’Gorman (1993) is based on a document spectrum (docstrum) for k-nearest-neighbor marks. In this approach, the first step is to determine the marks and their centroids. The direction vectors (distances, angles) for the k-nearest-neighbor connections of each mark are accumulated in a histogram. This histogram has two major peaks: one corresponding to intercharacter spacing, and the other corresponding to the document skew. Using these peaks, a complex process groups the characters into words and text lines. This method, including both skew estimation and layout analysis, is computationally expensive and ineffective in multi-size and short text strings. Using Gabor filters, which have been used earlier for the general problem of texture segmentation (Farrokhhina, 1990), Jain and Bhattacharjee (1992) propose a method for text segmentation of gray-level documents. The method works well with low-resolution documents and is robust to skew. Unfortunately, as the authors have reported, because of the frequency domain usage, the method is time-consuming, requiring about two minutes of a workstation CPU time for a $512 \times 512$ image. Finally, there are some methods that cannot be classified as either top-down or bottom-up. These methods are considered as hybrid. On this point, it is noted that the computational effort is a crucial criterion for the evaluation of document-segmentation techniques. Of course, the computational time must be significantly less than the time required if a character feature-extraction identification scheme is used.

This paper proposes an effective and fast bottom-up method that can classify and separate text from non-text regions for a wide class of documents. In addition to the identification of text regions, the proposed method can also identify line drawings and halftone regions. The method does not use morphological character features, and does not deal with the detailed extraction of isolated characters. The proposed technique is independent of the size and type of characters and the position of text and graphics in the document, and is tolerant to small skew errors. Its basic stages are:

- Identification of marks. Inclusion of each mark in a bounding box. Construction of the height histogram.
- Determination of the heights of the accepted boxes according to a histogram peak.
- Extension of boxes, and construction of bounding rectangles.
- Filtering of rectangles according to their base:height ratio.
- Texture feature extraction and classification.

The proposed method has two considerable advantages. The first is the substitution for the traditional and computationally expensive techniques for collinearity and vicinity checking of marks, by a new, simpler and faster one, which is based on box extensions. The other advantage is gained by the use of a new texture feature-analysis algorithm, which classifies the areas of the bounding rectangles as either text or non-text regions. The outcome of the method is the identification of areas containing only text. The remaining parts of the document might contain graphics and isolated characters.

The method was tested with many documents which contained text, line drawing and halftones. In this paper, typical examples are presented that cover special types of documents. The experimental results confirm the effectiveness of the proposed method, which has been implemented on a Pentium-100 PC using the C++ programming language. In this implementation, an average time of 5 sec per page of A4 size and 150 dpi resolution is required, without any effort to optimize the code for speed.
2. DESCRIPTION OF THE ALGORITHM

Usually, segmentation techniques tend to be iterative (O’Gorman, 1993) because they must segment documents with text of different sizes and types. The method described here for text classification of mixed-type documents, in each of its iterations, is based on the following general but reasonable assumptions about a text line:

(a) Text lines consist of characters (letters, numbers, symbols) of almost the same height, which are aligned in a straight line.
(b) The distance between neighboring characters of the same word is less than the size of their heights.
(c) If the characters in a text line are joined, then they can be enclosed in elongated bounding rectangles.

The number of necessary iterations is proportional to the main concentrations observed in the height histogram. These concentrations are obtained automatically by using a hill-clustering technique (Tsai and Chen, 1992; Papamarkos and Gatos, 1994) and correspond to characters of significantly different sizes. Capital letters, small letters, subscript and superscript type letters are examined together in the same iteration. Usually, at least four iterations are needed to cover the sizes of all the letters. The method is applied in documents having a binary form and, in summary, is composed of the following stages.

Step 1: In the original document, $I_1$, marks are extracted and identified, and then surrounded with bounding boxes. Next, a histogram is formulated from the heights of the bounding boxes.

Step 2: The histogram’s peaks are determined, using the hill-clustering method.

Step 3: For the iteration corresponding to a histogram peak value $H_{\text{max}}$, those boxes are accepted whose height $h$ satisfies the relation $H_{\text{max}}/2 \leq h \leq 2H_{\text{max}}$.

Step 4: The initial rectangular shape of each bounding box is extended to give a new image, $I_2$, which has touching boxes.

Step 5: The touching boxes derived in the previous step, are surrounded with bounding rectangles, and then a filtering is performed according to their base:height ratio.

Step 6: In the areas of $I_1$, defined by the bounding rectangles, a texture feature-extraction technique is applied, which then classifies them into four classes: first class of text, second class of text, class of halfones and class of drawings. The first class of text corresponds to normal-type characters, while the second text class includes mainly italic-type characters or numbers. Only areas in the first two classes are considered as text areas.

Step 7: For the next histogram peak with a smaller value than the previous one, Steps 3 through 6 are repeated until no other histogram peak exists.

These steps of the entire text-identification method are analyzed below.
2.1. Identification of marks and formulation of the height histogram

As defined in the introduction, a "mark" is a connected group of black pixels of the document image. Therefore, a mark is a set of pixels that includes all object pixels having at least one path leading to other pixels in the set. In a text region, a mark usually corresponds to a character, or to a disconnected part of a character. For effective mark determination, a fast contour following algorithm (CFA) is used, which is based on the extraction of the marks' boundary, without any morphological restriction (Pratt, 1991). All the stages of the segmentation method will be shown by applying it to the document of Fig. 1. This document is of low resolution (100 dpi), and is complex because it contains text of different sizes and types, text mixed with graphics, and graphics. Next, the height histogram of these boxes is formulated in the following way:

Suppose that there are \( N \) boxes, and each box \( i \) has height \( h_i \). Also, let \( H_j, j=1, 2, \ldots, K \) be the function giving the \( K \) different heights. The height histogram is derived by the relation:

\[
F(H_j) = \begin{cases} 
F(H_j) + 1, & \text{if } H_j = h_i \\
F(H_j), & \text{if } H_j \neq h_i
\end{cases}
\]

for \( i=0, 1, \ldots, N-1 \) and \( j=1, 2, \ldots, K \).
For the document of Fig. 1, Fig. 2 shows the height histogram.

2.2. Determination of the accepted boxes according to their heights

In each iteration of the method it is accepted that the characters have almost the same height in any text area. Taking advantage of this, the algorithm continues by finding those bounding boxes whose heights appear very often in the document. To accomplish this, the histogram described above is used. To determine the distributions and the peaks of the histogram, the well-known hill-clustering method (Tsai and Chen, 1992; Papamarkos and Gatos, 1994) is used. The result of this procedure is the determination of the histogram’s peaks, which correspond to the main distribution of heights in it. For a typical document, there is often a global peak for the distribution of characters of the predominant size, and, smaller peaks for the rest of the characters and noise. Therefore, $F(H)$ takes a global maximum value for the boxes that bound the characters of the most common size.

After the extraction of the histogram’s peaks, the algorithm proceeds iteratively in a peak-by-peak manner, starting from the biggest to the smallest peak, until no other peak exists. For each iteration, corresponding to the $H_{\text{max}}$ histogram peak value, only those boxes with heights $h_i$ satisfying the following condition are accepted

$$\frac{H_{\text{max}}}{2} \leq h_i \leq 2H_{\text{max}}$$  

(2)

The coefficients 1/2 and 2 have been obtained by examination of the relationship between the heights of the upper and lower-case characters. Similar coefficients are used in (Fletcher and Kasturi, 1988).

2.3. Extension and connection of the bounding boxes

After the construction and filtering of the bounding boxes, they are extended to give chains of connected boxes. This procedure is important for many reasons, mainly for the determination of the text lines. Also, this procedure gives independence from small skews in documents. Figure 3 shows the remaining boxes, after the height filtering and after having them extended by adding to their left and right sides two equal rectangular extensions, called “box-hands”. This procedure is depicted in Fig. 4. The value $h_i = H_{\text{max}}$ has been chosen appropriately to bring out the intersection of adjacent and extended boxes which are collinear and where the distance between them is less than $2H_{\text{max}}$. The values $h_1 = H_{\text{max}}/2$ and $h_2 = H_{\text{max}}/4$ have been chosen so that the boxes of tall characters (like $h$, $l$) or of characters with a tail (like $p$, $q$), can be joined with their adjacent boxes. Value $h_1$ is enough for the connection of bounding boxes of text lines with a small skew. A larger value of $h_1$ produces a larger tolerance to skew. This procedure results in the joining up of the bounding boxes of a text line, and the creation of chains of boxes which correspond to the text lines. In practice, this procedure is substituted for the complex and time-consuming use of the Hough Transform for collinearity and vicinity checking of the bounding boxes (Strouthopoulos et al., 1995).

2.4. Filtering of rectangles

Considering a new binary image $I_1$ as the result of the previous step, each mark (which now is composed of extended-connected boxes) is surrounded with new rectangles using the CFA. It is obvious that those rectangles that enclose extended boxes of a text line are elongated. In contrast, boxes that are located too far from their adjacent boxes, or boxes that constitute a small isolated group, either do not create overlapping boxes or create chains with relatively small lengths. Figure 5 shows the chains of the overlapping boxes, and the new rectangles which surround the chains. The new rectangles that have a base:height ratio greater than 3.5 are accepted. The threshold value 3.5 corresponds to the case with only two connected characters.

2.5. Texture classification

To classify the regions of the bounding rectangles as text or non-text areas, a texture-classification technique is applied. This technique is based on 13 powerful structural features which are called document structure elements (DSE). A DSE is any $3 \times 3$ binary block. The order of the pixels of such a block is shown in Fig. 6.

Assign to any DSE an integer $L = \sum_{i=0}^{5} b_i2^i$, and call $L$ the DSE characteristic number (DSECN). It is obvious that since $L \in \{0, 1, 2, \ldots, 511\}$, there are $2^6 = 512$ different blocks of DSEs, and there is a one-to-one correspondence between DSEs and their DSECNs. For a rectangular area $A$, let $K$ be the number of columns and $J$ the number of rows. It is obvious that $A$ has $(K-2)(J-2)$ DSEs, and for any $i$ DSE there is a characteristic number $\ell_i$. The DSEs histogram function $G(L)$ of area $A$ is derived by the relation

$$G(L) = \begin{cases} 
G(L) + 1, & \text{if } \ell_i = L \\
G(L), & \text{if } \ell_i \neq L 
\end{cases}$$

(3)

for $i = 0, 1, 2, \ldots, (K-2)(J-2) - 1$ and $\ell_i \in \{1, 2, \ldots, 510\}$. Note that DSEs 0 and 511 are not considered because they correspond to pure background and object regions, respectively.

For this histogram, the probability density function $S(L)$ is equal to
Fig. 7. Probability density functions $S(L)$ of typical regions.
\[ S(L) = \frac{G(L)}{\sum_{L=1}^{10} G(L)} \]  \hspace{1cm} (4)

Figure 7 shows the \( S(L) \) functions of the four typical regions, i.e., regions corresponding to the first class of text, the second class of text, drawings and halftones.

The 510 values of \( S(L) \) can be taken as texture features. However, as is explained below, by the application of a feature-reduction technique, only 13 of them are finally selected as texture features.

In the classification stage, these 13 features are used in combination with a minimum distance classifier having only four classes: text of normal characters, italic text, drawings and halftones. Specifically, if \( i = 1, 2, 3, 4 \) and \( j = 1, 2, \ldots, 13 \) are indexes of the number of classes and the number of features, respectively, the four distances \( D_i \) are estimated by the relation

\[ D_i = \sqrt{\sum_{j=1}^{13} \left( \frac{f_{ij} - \mu_{ij}}{\sigma_{ij}} \right)^2} \]

where \( f_{1j}, f_{2j}, \ldots, f_{13j} \) are the features of region \( A, \mu_{1j}, \mu_{2j}, \ldots, \mu_{13j} \) are the coordinates of the center of class \( i \) and \( \sigma_{1j}, \sigma_{2j}, \ldots, \sigma_{13j} \) are the variances of class \( i \).

Let \( D = \min \{ D_1, D_2, D_3, D_4 \} \). If \( D < 1 \) and \( D = D_1 \) or \( D = D_2 \), then \( A \) is classified as a text area; otherwise, \( A \) is a non-text area. Table 1 gives the thirteen features, the DSECN, \( L \), the coordinates of each class center in the 13-dimensional feature space, the shape of each DSE and the classes containing the maximum and minimum number of DSE.

### 2.6. Feature reduction and clustering

An important process in a recognition system is to select a small set of appropriate features from a much bigger set. Selection of "good" features is critical to the performance of classification. To achieve this, a feature-reduction procedure was applied that permits the selection of only 13 of the 512 features. To reduce the number of 512 features, selecting a small set of "good" features which allow a region to be correctly classified as text, drawings, or halftones, a large number of such regions was examined, and an evaluation process was performed which examined the stability, separability and similarity of the features.

For the stability test, an attempt was made to identify the features which give significantly stable performance. The

<table>
<thead>
<tr>
<th>Feature</th>
<th>DSECN</th>
<th>Centers of classes</th>
<th>DSE</th>
<th>MAX</th>
<th>MIN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1st cl. of text</td>
<td>2nd cl. of text</td>
<td>Halftones</td>
<td>Drawings</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>219</td>
<td>0.062</td>
<td>0.0393</td>
<td>0.0011</td>
<td>0.0011</td>
</tr>
<tr>
<td>02</td>
<td>73</td>
<td>0.062</td>
<td>0.0385</td>
<td>0.0015</td>
<td>0.071</td>
</tr>
<tr>
<td>03</td>
<td>438</td>
<td>0.0614</td>
<td>0.0354</td>
<td>0.0025</td>
<td>0.0011</td>
</tr>
<tr>
<td>04</td>
<td>292</td>
<td>0.06</td>
<td>0.0345</td>
<td>0.0027</td>
<td>0.0072</td>
</tr>
<tr>
<td>05</td>
<td>1</td>
<td>0.0251</td>
<td>0.0473</td>
<td>0.013</td>
<td>0.0345</td>
</tr>
<tr>
<td>06</td>
<td>256</td>
<td>0.0232</td>
<td>0.0454</td>
<td>0.0128</td>
<td>0.0344</td>
</tr>
<tr>
<td>07</td>
<td>170</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0467</td>
<td>0.0</td>
</tr>
<tr>
<td>08</td>
<td>341</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0425</td>
<td>0.0</td>
</tr>
<tr>
<td>09</td>
<td>186</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0264</td>
<td>0.0016</td>
</tr>
<tr>
<td>10</td>
<td>495</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0243</td>
<td>0.0</td>
</tr>
<tr>
<td>11</td>
<td>448</td>
<td>0.038</td>
<td>0.029</td>
<td>0.0025</td>
<td>0.0917</td>
</tr>
<tr>
<td>12</td>
<td>7</td>
<td>0.04</td>
<td>0.023</td>
<td>0.0047</td>
<td>0.0917</td>
</tr>
<tr>
<td>13</td>
<td>56</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0001</td>
<td>0.0737</td>
</tr>
</tbody>
</table>
For the feature similarity analysis, and for every two features \( \ell \) and \( m \) belonging to the same class \( p \), the correlation factor

\[
C_{\ell m} = \frac{1}{n_p} \sum_{i=1}^{n_p} (\ell_i - \mu_\ell)(m_i - \mu_m)
\]

was estimated, where \( n_p \) is the number of elements of class \( p \), \( \ell \) and \( m \) the feature values of element \( i \), \( \mu_\ell \), \( \mu_m \), \( \sigma_\ell \) and \( \sigma_m \) the means and standard deviations of the features in the class \( p \). The correlation factor measures the similarity between two features, and gives values between \(-1\) and \(+1\). A value near \(-1\) or \(+1\) means that the two features are highly correlated or inversely correlated, respectively. A value near zero indicates that the features are highly uncorrelated. Features with \( |C_{\ell m}| > 0.9 \) must be rejected, and it was found that 49 features could be rejected.

After the application of the above three feature-reduction tests, only 13 features remained having the ability to classify the regions into classes. For classification, the Nearest Means Clustering (NMC) algorithm (Coleman and Andrews, 1979) is used which can determine the possible classes and their centers. The NMC algorithm has been applied to a large number of samples, and it has been found that except for the initial three classes (normal type text, drawings and halftones) there is also a fourth class, corresponding to regions of italic characters and numbers.
13.4 INITIALIZATION

The primitive equations of the atmospheric flow are solved, generally subject to the initial and boundary conditions, usually based on past observations. The analysis step is essential for initializing the model, providing the necessary conditions for the model to produce meaningful forecasts. This step involves the assimilation of observational data into the model fields, typically using a data assimilation system. The process aims to compute the model state at a specific time, given the observations and the model's initial conditions. The analysis step can be performed using various methods, such as optimal interpolation, 3D-Var, or 4D-Var techniques. These methods combine the model fields with the observations to produce an initialized state that is used as the initial condition for the next forecast cycle. For this reason, the analysis step is often performed offline, separate from the analysis with the atmospheric model, and the solutions produced by these methods are then incorporated into the operational analysis. The initialization step ensures that the model state is consistent with the observations, minimizing the model's error and improving forecast accuracy.

14.1 NWP PRODUCTS APPLIED TO AVIATION

World Area Forecasts. These are required to distribute grid-point data of wind and temperature at various levels, as well as information on the tropopause and the maximum wind. Suitable degraded in graphical or chart form, these data alone are of great value to aviation forecaster. However, a wide range of ancillary fields and derived data may be generated from an NWP system which provide a more complete picture of the numerical forecast. HR-12 forecasts from the ECMWF operational models are used to illustrate the range of products which can be processed from a numerical system. Most are used routinely at Handsworth by the Regional Area Forecast Centre. Some are derived from fields issued nationally on the GTS, but many require data of a format only available at the centre. Assessments of data from the ECMWF and other global models is shown here, but all products could equally well have been derived from a high-resolution global model. In all cases, the model products have been projected onto a uniform grid (about 100 km grid spacing) for output purposes, which is considerably finer than the resolution of data available on the GTS in 1990. The output resolution, both in the horizontal and vertical, of course, greatly affects the detail that can be identified.

Fig. 9. (a)-(c) Typical examples.
VORTEX™
Concentrates I/O
for top C40
performance

Harness the enormous power of TMS320C40 digital signal processors in a small footprint – the Vortex system.

The Vortex C140 Concentrator provides an interface for processor-to-processor communication among any number of processor boards. This Concentrator can be assembled quickly and easily by plugging these processor boards into the Vortex Concentrator. The Vortex Concentrator system is ideal for any high-speed processor-to-processor applications.

The Vortex Concentrator can be used with the C40 for I/O processing, or for entirely separate functions. This gives the Vortex a total floating-point capacity of 85 MFLOPS.

The Vortex Concentrator accommodates up to 2 Mbytes of SRAM and works with ASPI's digital signal processor boards. Available are 4 Mbytes of DRAM boards, digital audio interfaces, and general A/D/D/A systems. Ask today for full details on this remarkable new DSP system.

Fig. 9. Continued.
This class was named the second class of text. It is important to note that the objective here is to identify regions that can be successfully classified as text. Consequently, regions belonging to this new class are also labeled as text. Table 1 summarizes the results of the application of the feature-reduction and clustering procedure. As can be observed from this table, these results are reasonable and expected. For example, DSEs with characteristic numbers 170, 341, 186, 495 are the main components of halftone pictures.

Figure 8 shows the final separation results for the document of Fig. 1. It can be observed from the bottom part of this figure that good separation results are achieved, even for worse text regions.

3. EXPERIMENTAL RESULTS

The proposed method was extensively tested on a series of mixed-type documents. In addition to the authors' own test documents, the method has also been applied on some documents obtained from the University of Washington database (UW, 1993). The results appear to be very promising. To demonstrate the effectiveness of the proposed segmentation method some additional typical examples are presented.

Figure 9 presents the results obtained by applying the proposed method to six mixed-type documents. The document of Fig. 9(a) is of size 1688 x 2200 pixels and 300 dpi resolution, and is taken from the University of Washington database. Figure 9(b) and (c) are selected mixed-type documents with different types of segmentation difficulties. These documents were scanned by a HP ScanJet IIc scanner at a 150 dpi resolution. All the documents in Fig. 9 contain text with fonts of different types, styles and sizes, and graphics that cannot be separated using vertical and horizontal lines. As can be observed, accurate text identification results were obtained.

As a final example, the proposed method was applied on two documents which have been artificially skewed by 5° and 10°. Figures 10(a) and (b) show the text identification results, respectively. It is noticed that the suggested method works well, even in the case of Fig. 10(b), which is a complex document that contains text and straight lines crossing each other.

4. CONCLUSIONS

Segmentation is an important issue in the automated document analysis research area. Text segmentation plays a significant role in document retrieval and storage systems. This paper proposes a new segmentation method that clusters the regions of a mixed-type document into text or non-text areas. The method belongs to the bottom-up class, and consists of the following main stages: extraction of marks, filtering of marks according to their heights, extension of mark shapes and creation of chains of connected marks, surrounding of chains by bounding rectangles, and finally, clustering the regions of the bounding rectangles as text or non-text areas. For the final stage, a technique is applied which is based on document
structure elements, and which formulates textural features. Specifically, 510 textural features were initially considered. Then, by using a feature-reduction process, only 13 of them were kept. Each of the 510 features corresponds to the binary contents of a 3 × 3 mask, and its value is equal to the frequency distribution of the mask in the examination area. The application, in the training stage, of a clustering technique results in four classes: first class of text, second

Fig. 10(a) Results for a document skewed by 5°.
class of text, class of halftones and class of drawings.

In comparison with other techniques, this method has two advantages. First, for the mark-filtering stage a fast algorithm was developed and used for collinearity and vicinity checking. Second, the final stage of the method is based on the use of a texture feature analysis algorithm which can easily classify the areas of the bounding rectangles as text or non-text regions.

The proposed segmentation method was extensively tested on many mixed-type documents having significant difficulties for segmentation. It is independent of the size and type of the characters and the position of the text in the document, and it is insensitive to small tilts. It works well even in the cases where graphics and text areas cannot be separated by vertical and orthogonal lines. Also, the method can separate out any text included in graphic regions or in mathematical equations. The results were very promising, and under normal scanning conditions no wrong classifications of a region as a text area were found. The average time for processing of a page is about 5 sec, on a Pentium-100 computer.

REFERENCES
Farrokhnia, F. (1990) Multi-channel filtering techniques for texture
C. STROUTHOPOULOS et al.: TEXT-ONLY AREAS

Stroutthropoulos, C., Papamarkos, N. and Chamzas, C. (1995) Identification of text-only areas in mixed type documents. *IEEE Workshop on Non-
linear Signal and Image Processing*, 1, 162–165.

AUTHORS’ BIOGRAPHIES

Charalampos Stroutthropoulos was born in Drama, Greece, in 1962. He received his Diploma Degree in Electrical Engineering from the University of Patras, in 1985. Since 1993 he has been a Ph.D. student in the Electrical and Computer Engineering Department of Democritus University of Thrace, in the field of image segmentation and recognition. His current interests include document analysis and image processing. C. Stroutthropoulos 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 Assistant Professor at the Democritus University of Thrace where he is currently an Associate Professor (since 1996). His current research interests are in digital signal processing, filter design, image processing, pattern recognition and computer vision. Dr Nikos 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), and a member of the Visual Communications Research Department, AT&T Bell Laboratories (1982–1990) and is now an Associate Professor at the Democritus University of Thrace (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.