Journal of Computer & Robotics 1 (2008) 9-23

Computer
& Robotics

Persian Printed Document Analysis and Page Segmentation

Ali Broumandnia[a,*], Jamshid Shanbehzadeh[b]

[a]*Department of Computer & IT, Islamic Azad University-South Tehran Branch, Tehran, Iran*
[b]*Department of Computer, Tarbiat Moalem University, Iran*

Received 2007 July; revised 2008 March; accepted 2008 April

Abstract

This paper presents, a hybrid method, low-resolution and high-resolution, for Persian page segmentation. In the low-resolution page segmentation, a pyramidal image structure is constructed for multiscale analysis and segments document image to a set of regions. By high-resolution page segmentation, by connected components analysis, each region is segmented to homogeneous regions and identifying them as texts, images, and tables/drawings. The proposed method was experiment with the Persian documents. The result of these tests have shown that the proposed method provide more accurate and speed results.

Keywords：Page segmentation, pyramidal image structure, connected components, horizontal and vertical merging.

1. Introduction

With the advances in communication and information technology in today's world, the volume of automated processing and reading of documents has increase more and more. In spite of use of electronic documents, the amount of hardcopy documents has never decreased because most human beings prefer hardcopy documents for reading and archiving. Therefore efforts have been made to store the hardcopy documents in digitized format, but that require an enormous storing space, even after compression using modern techniques. Furthermore, access and transferring of digitized documents consumes too much time. If hardcopy documents could be converted to electronic format, it would be possible to remote transferring for the contents of documents in seconds and to access and update them efficiently. By separating the text and the graphic/image part of hardcopy document and storing text as a character set and the graphics/image part as bitmap, the hardcopy document are converted to electronic forms.

The procedure of converted hardcopy documents to electronic documents is called document image analysis. The document image analysis includes page segmentation, optical character recognition, image compression, table reconstruction and drawing vectoring. The goal of page segmentation is to segment a document image into homogeneous regions such as text regions, images regions, table regions and drawing regions. This paper focused on the Persian page segmentation.

A number of approaches have been proposed for page segmentation.

The approaches for page segmentation are typically referred to as bottom-up and top-down methods [1], [2], [3], [4], [5], [6], [7]. The bottom-up methods start pixels or the connected components, determine the words, merge the words into text lines, and merge the text lines into paragraphs. A disadvantage of these approaches is that the identification, analysis, and grouping of connected components are, in general, time-consuming processes, especially when there are many components in the image. The top-down approaches look for global information on the page, e.g., black and white stripes, and on the basis of this, split the page into columns, the columns into blocks,

* Corresponding author. Tel.: +98-912-3430074; fax: +98-21-22377001; e-mail: broumandnia@{gmail.com, azad.ac.ir}.

the blocks into text lines, and the text lines into words. The advantages of these approaches are that the time complexity is lower than that of bottom-up approaches and it is natural for human beings to see an object from a coarse to fine resolution. However, with the previous methods, the complex document layout which is composed of nonrectangular images and various character font sizes makes it difficult to segment correctly in a top-down manner.

Some of page segmentation methods regard a homogeneous region such as text, image or graphic, in a document image as a textured region. Then, page segmentation is implemented by finding textured regions in gray scale images. One major problem associated with such texture-based approaches [8], [9], [10] is that the time complexity is too high since different filters are tuned to capture a desired local spatial frequency and the orientation characteristics of a textured region, so that many masks are used for extracting local features.

Since the Persian documents have some special characterize compared with the English documents, then can not be directly used the above methods for Persian page segmentation. The special characterize of Persian documents as follow:

1. The Persian scripts are cursive and each connected components include more than one character. The arrangement and the size of these connected components also vary greatly.
2. The Persian alphabet, there are 32 basic characters. These characters may change their shapes according to their positions (beginning, medium, end or isolated) in the word. Each character can take up to four different shapes, as result we have 114 different shape for all of Persian alphabets. The use of special stress marks called dots is another characteristic of Persian scripts. Most of the Persian characters have one, two or three dots. These dots can be situated at the top, inside or bottom of the character s. An important feature of this script, from the script identification point view, is the non uniformity of words size. The word size varies according to number of cursive characters in word.

In this paper, we propose a hybrid method, low-resolution and high-resolution page segmentation, to Persian page segmentation. In the low resolution page segmentation, we construct a pyramidal image structure and segmented Persian documents into a set of regions. After low-resolution, in the high resolution page segmentation each regions are segmented to homogenous regions. The connected components analyses are used in high-resolution page segmentation.

This paper is organized as follows: In Section 2, connected components analysis with propose algorithms are described. In Section 3, pyramidal image structure for low-resolution analysis is introduced. In section 4, a hybrid method for Persian page segmentation and region identification is proposed. In Section 5, to verify the performance of the proposed method, experimental results are analyzed. Finally, conclusions and further research directions are given in Section 6.

2. Connected Component Analysis (CCA)

The concepts of connectivity and connected components were introduced in references [11], [12]. In practice, extraction of connected components in a binary image is central to many document image analysis applications. Let C_i represent a connected components contained in an image P and assume that a point of C_i is known. Then following morphology iterative expiration yield all the element of P:

$$X_k = (X_{k-1} \oplus B) \bigcap P \qquad k = 1,2,3 \qquad (1)$$

The Where $X_0 = P$, $X_{k-1} \oplus B$ is the dilation of X_{k-1} by B and B is a 3×3 strutting elements of 1's.

If $X_k = X_{k-1}$, the algorithm has converged and we let $C_i = X_k$.

Note that the shape of the structuring elements assumes 8-connectivety between pixels and assuming that a point is known in each connected component.

Based on a connected components analysis approach, we segmented image pixels of P into connected components, connected component into bounding box, bounding box into graphic text lines (GTLs) and GTLs into region blocks. In the following section, we described connected component analysis more details.

2.1. Bounding Box

We represent the binary image of a page P as a connected components analysis (CCA) which is defined as $C = \{C_i\}$, where $C = \{C_i\}$ is a set of connected components [13]. Each connected components C_i is characterized by its upper left and lower right rectangular coordinates $((X_u(C_i), Y_u(C_i)), (X_l(C_i), Y_l(C_i)))$, where $X_u(C_i) < X_l(C_i)$ and $Y_u(C_i) < Y_l(C_i)$ .The connected components extracted from Fig. 1(a) are shown with their bounding boxes in Fig. 1(b). Some very small connected components and those close to the boundaries of the image are treated as noise and deleted..

2.2. Bounding Boxes Distances

At The definition of distance between any two bounding boxes proposed by Simon[3] for component classification and clustering. The distance between any two connected components C_i and C_j or two objects O_i and O_j is expressed as distance between their bounding boxes. Let us

define a general object O_i which can be a connected component C_i, a GTL t_i, or a region block b_i, in terms of its bounding boxes with left, right, top, and bottom coordinates denoted as $((X_u(O_i), Y_u(O_i)), (X_l(O_i), Y_l(O_i)))$, where $X_u(O_i) < X_l(O_i)$ and $Y_u(O_i) < Y_l(O_i)$. We define the width $W = X_l - X_u$ and height $H = Y_l - Y_u$ of an object as its horizontal and vertical extent, respectively. Referring to Fig. 2 the horizontal and vertical distances between two objects are defined as

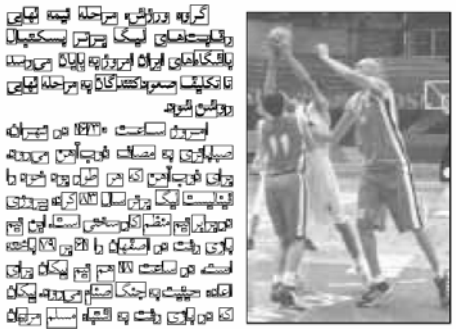
$$D_x(O_i, O_j) = \max[X_u(O_i), X_u(O_j)] - \min[X_l(O_i), X_l(O_j)] \quad (2)$$

and

$$D_y(O_i, O_j) = \max[Y_u(O_i), Y_u(O_j)] - \min[Y_l(O_i), Y_l(O_j)] \quad (3)$$

respectively. Note that if $D_x(O_i, O_j) < 0$, then objects O_i and O_j overlap in horizontal direction. We can define the vertical overlap between two objects in an analogous fashion.

a) Persian documents image



b) Bounding boxes of connected components

Fig. 1. Connected components analysis and bounding box

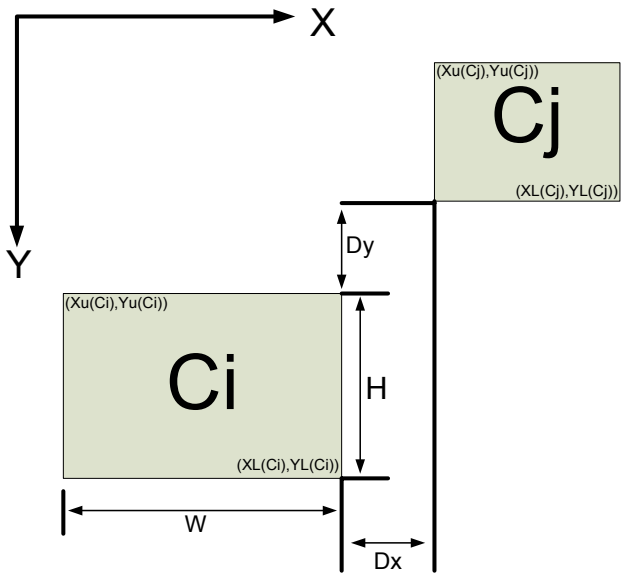


Fig. 2. The horizontal and vertical distance of two objects

2.3. Horizontal Merging of components

In the horizontal merging, we groups connected components, $C = \{C_i\}$, into GTLs. For given a set of connected components $C = \{C_i\}$, we say any two C_i and C_j are horizontal close and vertically overlapped at distance T_d, if

$$D_x(C_i, C_j) < T_d, \qquad V_y(C_i, C_j) < 0.25 \quad (4)$$

Where

$$V_y(C_i, C_j) = \frac{-D_y(C_i, C_j)}{\min[H(C_i), H(C_j)]} \quad (5)$$

Where

$$H(C_i) = Y_l(C_i) - Y_u(C_i) \quad (6)$$

$$H(C_j) = Y_l(C_j) - Y_u(C_j) \quad (7)$$

If two connected components, C_i and C_j, are satisfied in equation (4), where $T_d = 10$. Then C_i and C_j are merged and the upper-left and lower-right coordinates are calculated as follow:

$$X_u = \min[X_u(c_i), X_u(c_j)] \quad , \qquad (8)$$
$$X_l = \max[X_l(c_i), X_l(c_j)]$$

$$Y_u = \min[Y_u(c_i), Y_u(c_j)] \quad , \qquad (9)$$
$$Y_l = \max[Y_l(c_i), Y_l(c_j)]$$

For merging, the bounding boxes of connected components are stored in a linked list. Therefore each element of linked list contains coordinates of a bounding box. If C denoted linked list of bounding boxes, using the inter component distance defined in (4), we can merge connected components, $C = \{C_i\}$, into GTLs by following algorithms (Fig. 3):

1. Copy C linked list to T linked list.
2. For all pairs of T elements, t_i and t_j, which satisfied in (4) do following operations.
Merge t_i and t_j to t_m by equations (8) and (9).
Remove t_i and t_j from T linked list.
Insert t_m to T linked list.
3. Repeat step (2) until all pairs of T elements dissatisfied in (4).
4. The final T linked list is GTLs.

Fig. 3. Horizontal merging of bounding boxes

2.4. Vertical Merging of bounding boxes

In the vertical merging, we groups some bounding boxes, $B = \{b_i\}$. For given a set of bounding boxes, $B = \{b_i\}$, we say any two b_i and b_j are vertical close and horizontally overlapped at distance T_b, if

$$D_y(b_i, b_j) < T_b,$$
$$V_x(C_i, C_j) < 0.5, \qquad (10)$$
$$0.9 \le X_l(b_i) / X_l(b_j) \le 1.2$$

Where

$$V_x(b_i, b_j) = \frac{-D_x(b_i, b_j)}{\min[W(b_i), W(b_j)]} \qquad (11)$$

Where

$$W(b_i) = X_l(b_i) - X_u(b_i) \qquad (12)$$

$$W(b_j) = X_l(b_j) - X_u(b_j) \qquad (13)$$

If two bounding boxes, b_i and b_j, are satisfied in equation (11), where $T_b = 20$. Then b_i and b_j are merged and the upper-left and lower-right coordinates are calculated as follow:

$$X_u = \min[X_u(b_i), X_u(b_j)] \quad , \qquad (14)$$
$$X_l = \max[X_l(b_i), X_l(b_j)]$$

$$Y_u = \min[Y_u(b_i), Y_u(b_j)] \quad , \qquad (15)$$
$$Y_l = \max[Y_l(b_i), Y_l(b_j)]$$

For merging, the bounding boxes of $B = \{b_i\}$ are stored in a linked list. Therefore each element of linked list contains coordinates of a bounding box. If T denoted linked list of bounding boxes, using the inter bounding boxes distance defined in (10), we can merge bounding boxes, $B = \{b_i\}$, by following algorithms (Fig. 4.):

I. 1. Copy T linked list to B linked list.
II. 2. For all pairs of B elements, b_i and b_i, which satisfied in (10) do following operations.
III. Merge b_i and b_i to b_m by equations (14) and (15).
IV. Remove b_i and b_i from B linked list.
V. Insert b_m to B linked list.
VI. 3. Repeat step (2) until all pairs of T elements dissatisfied in (10).
VII. 4. The final B linked list is result of vertical merging.

Fig. 4. Vertical merging of bounding boxes

Ensure that all tables, figures and schemes are cited in the text in numerical order. It is strongly recommended that authors follow the recommendations of the IUPAC Manual of Symbols and Terminology for Physico-chemical Quantities and Units, edited by IM Mills, Blackwell, Oxford, 1988. All measurements should be given in the form consistent with "quantity calculus" (see J. Electroanal. Chem. 271 (1989) 370). Abbreviations should be used consistently throughout the text, and all nonstandard abbreviations should be defined on first usage. The experimental information should be as concise as possible, while containing all the information necessary to guarantee reproducibility.

3. Pyramidal Image Structure

A powerful, but conceptually simple structure for representing images and more than one resolution is the image pyramid [11]. An image pyramid is a collection of decreasing resolution images arranged in the shape of pyramid. As can be seen in Fig. 5 the base of the pyramid contains a high-resolution representation of the image being processed; the apex contains a low resolution approximation. Fully populated most pyramids are composed of all resolution levels, but most pyramids are

truncated to P levels. The pyramid image is made by reducing the resolution of an image with size $M \times N$ by quarter repeatedly while its size is larger than $M' \times N'$, where M' or N' is less than 100.

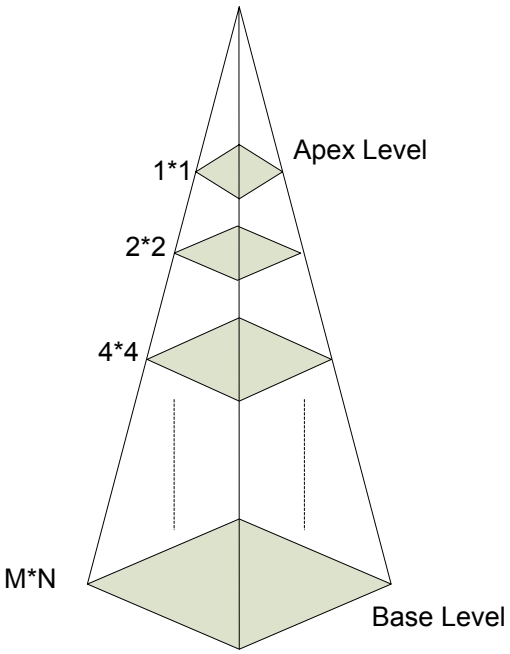


Fig. 5. Pyramidal Image Structure

Fig. 6 shows a simple system for constructing image pyramids. The level I_{i+1} approximation output is used to create approximation pyramids, which contain one or more approximation of the original image. As can be seen in Fig. 6 OR logic are first applied in pairs of adjacency odd and even columns in image I_i and then OR logic are applied in pairs of adjacency odd and even rows in image I_{i3}. Therefore, the number of pixels in I_{i+1} is on quarter of the number of pixels in I_i. As a result of this procedure, the multiscale images shown in Fig. 5 are obtained. Executing this procedure L times two intimately related L+1 level approximation pyramids. The level L approximation outputs are used to popular the approximation pyramids.

4. Proposed Page Segmentation Method

Fig. 7 shows a block diagram for proposed page segmentation. The proposed method for page segmentation and region identification are described by algorithm of Fig. 8.

4.1. Low-resolution page segmentation

In multi resolution analysis, a pyramid's lower-resolution levels can be used the analysis of large structure or overall image context. In this stage with low-resolution image approximation, we segment binary document image P_0 in n individual regions.

Let P_L denote the low-resolution image in pyramidal image structure. A connected components analysis is applied to the foreground regions of P_L to produce the set of connected components at low-resolution of pyramidal image structure. Then, for each connected component, its associated bounding boxes – the smallest rectangular box which circumscribes the component-is calculated. The bounding boxes of P_L connected components are represented by giving the coordinates of the upper left and the lower right corners of the box.

Fig. 9(a) is shown a Persian document image and Fig. 9(b) is shown the bounding boxes in low-resolution image of Fig. 9(a).

The bounding boxes of connected components in image P_L specify n individual regions. These regions denote by R_i $(i = 1,2,....,n)$ and define as follow:

$$\begin{aligned} &((X_l^{P_L}(R_i),(Y_l^{P_L}(R_i)),\\ &((X_u^{P_L}(R_i),(Y_u^{P_L}(R_i))\\ &i = 1,2,.....,n \end{aligned} \tag{16}$$

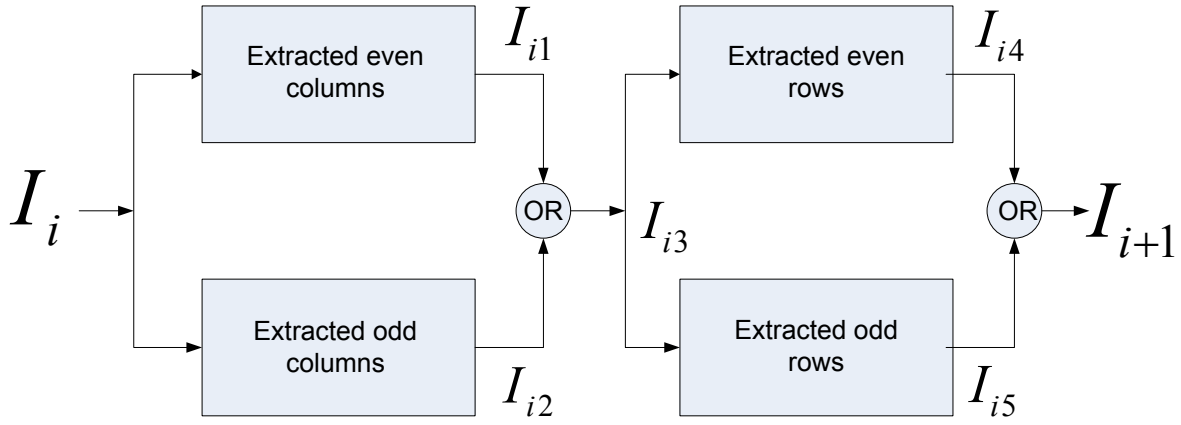


Fig. 6. The simple system for constructing one stage of image pyramids

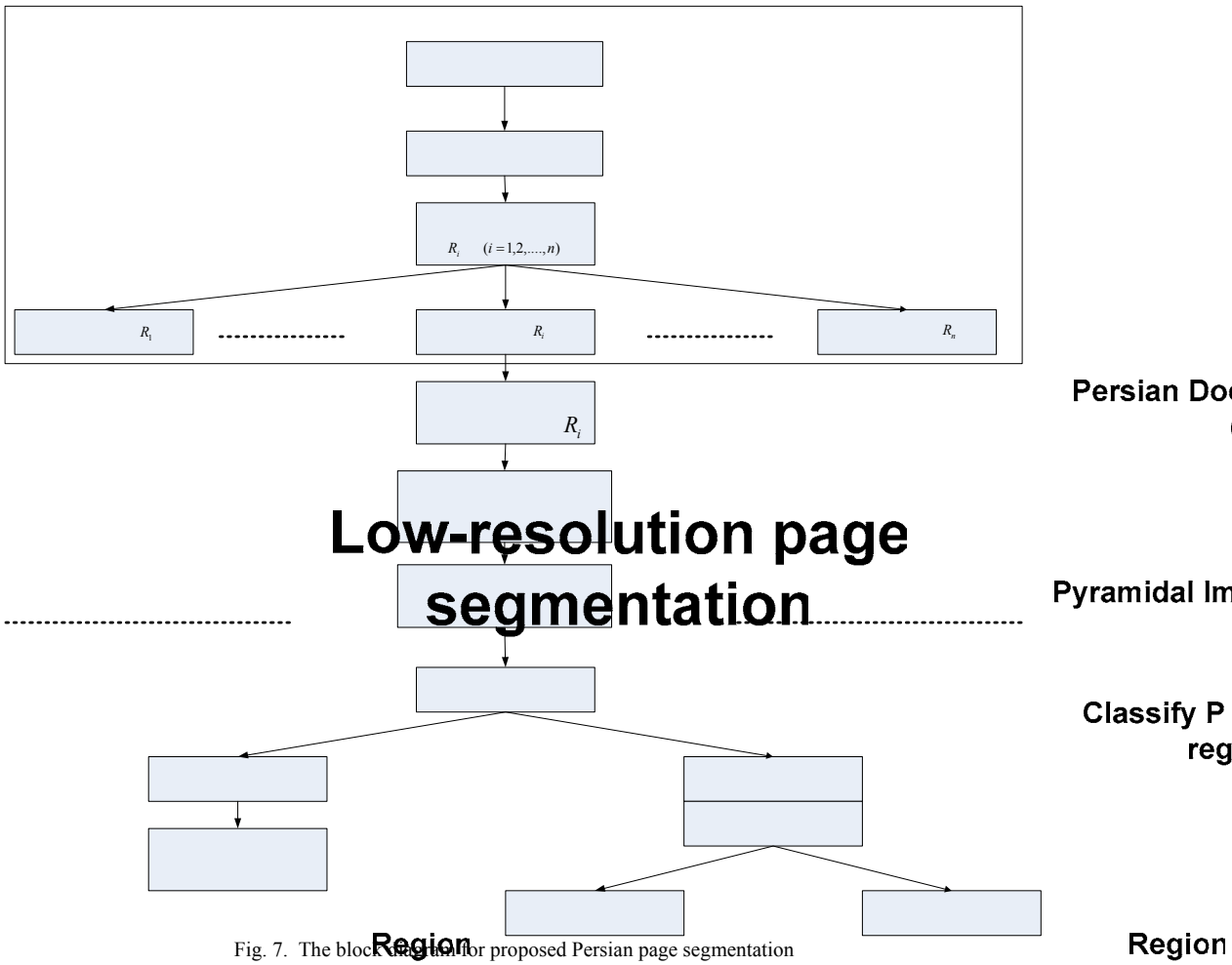


Fig. 7. The block diagram for proposed Persian page segmentation

Low-resolution page segmentation

1. Construct a pyramidal image structure and get low-resolution image P_L of the input document image P_0 (section 3).
2. Extract bounding boxes of connected components for the low-resolution image P_L (section 4.1).
3. Classify binary document image P_0 to set of regions R_i (i=1,2,….,n) (section 4.1).

High-resolution page segmentation

4. For all regions R_i (i=1,2,….,n) do following steps (section 4.2).

Skew correction.
Remove overlapped connected components.
Obtain connected components.
Horizontal merging and get GTLs.
GTLs classification into text lines (TLs) and non text lines (NTLs).
Vertical merging of TLs.
NTLs classification into images and drawings or graphics table.

Fig. 8. Proposed Persian page segmentation method

As can be seen in Fig. 9, if pyramidal image structure contains L low-resolution approximation. The coordinates of bounding boxes $R_i \quad (i=1,2,....,n)$ in base image P_0 are calculated as follow:

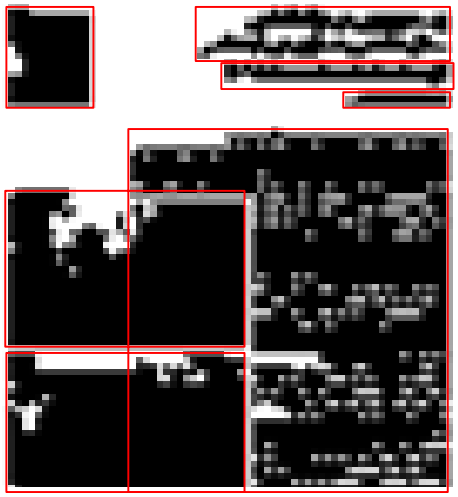
$$\begin{aligned}
&((X_l^{P_0}(R_i),(Y_l^{P_0}(R_i))= \\
&((X_l^{P_L}(R_i),(Y_l^{P_L}(R_i))\times 2^L \\
&((X_u^{P_0}(R_i),(Y_u^{P_0}(R_i))= \\
&((X_u^{P_L}(R_i),(Y_u^{P_L}(R_i))\times 2^L \\
&i=1,2,.....,n
\end{aligned} \tag{17}$$

Where P_0 and P_L are base image and low-resolution image in level L respectively.

4.2. High-resolution page segmentation

In the high-resolution page segmentation, all of regions $R_i \quad (i=1,2,....,n)$ in base image P_0 are segmented into text, image, drawing and table regions. Its high-resolution image in pyramidal image structure is appropriate for analyzing individual object characteristic. Such as coarse to fine analysis strategy is particularly useful in pattern recognition and page segmentation. The high- resolution page segmentation is described in this section.

a) The Persian document image



b)The bounding boxes of connected components in low-resolution (top of pyramidal image structure)

Fig. 9. The Persian document image in base and top of pyramidal structure

4.2.1. Skew correction

The skew angle estimation and correction of a document page is an important task for document analysis and optical character recognition.

There are several popular approaches for skew detection [14]. Some approaches worked on the pixel level, such as those using projection profiles [15]; others work on the connected components level such as those using Hough transform [16] and nearest- neighbourhood [17].

For skew correction of regions, $R_i \quad (i=1,2,....,n)$, we used proposed method in reference [18].

4.2.2. Remove overlaps connected components

In the low-resolution page segmentation, the binary image of document P_0 is divided into a set of regions that called $R_1, R_2, ..., R_n$. Each region R_i is defined by coordinates of bounding box. Some of regions, R_i, are overlapped to other. For next stage of high-resolution page segmentation, we should remove overlapped components form each region R_i. For each region, we remove overlapping components as follow:

First, we obtained connected components of region R_i in low-resolution image P_L. If R_i have only one connected component, then R_i do not have any overlapping

components. If in region R_i number of connected components is more than one, we leave the maximum of connected component and remove other connected components at the high-resolution. Fig. 10(a) shows all regions R_i for sample document image in the low-resolution and high-resolution. Fig. 10(b) shows regions R_i after we removed overlapping components.

a) The base and top image in pyramidal structure for a sample Persian documents

b) Four regions after removing overlaps components

Fig. 10. The low-resolution segmentation and removing overlap components

4.2.3. Obtain connected components

For high-resolution page segmentation of region R_i, connected components are extracted by equation (1). The specifications of R_i connected components are specified by coordinates of bounding boxes. These specifications are stored in a desirable linked list for next stages. Fig. 11 is shown connected components of a region.

Fig. 11. The connected components of a region

4.2.4. Horizontal Merging and get GTLs

This stage are described more detail in section 2.3. In this stage, connected components of region R_i are divided into Graphic Text Lines (GTLs) by algorithm of Fig. 3. However, Fig. 12 is shown result of horizontal merging on regions Fig. 11.

4.2.5. GTLs classification

Both handwritten and printed Persian scripts are cursive. The shape of a Persian character is a function of its location a word, where each character can have two to four different forms. Most characters have one, two, or three dots which can be located above, bellow, or inside a character. As result connected components of GTLs have different width and height sizes so connected components analysis not suitable for GTLs classification.

Persian text line region can be easily distinguished from other regions by property that horizontal projection of a text line has exactly one peak and vertical projection have at least two peaks and one valley. If we apply this property of text lines regions to GTLs, we could easily classify GTLs regions.

In order to classify a GTL, we first obtain the vertical and horizontal profiles, by equations (18) and (19). In these equations, $GTL^i(x,y)$

$$P_V^i = \{P_x \mid P_x = \frac{1}{H}\sum_{y=0}^{H-1} GTL^i(x,y), \quad 0 \le x < W \} \tag{18}$$

$$P_H^i = \{P_y \mid P_y = \frac{1}{W}\sum_{x=0}^{W-1} GTL^i(x,y), \quad 0 \le y < H \} \tag{19}$$

Where P_V^i and P_H^i are normalized between zero to one.

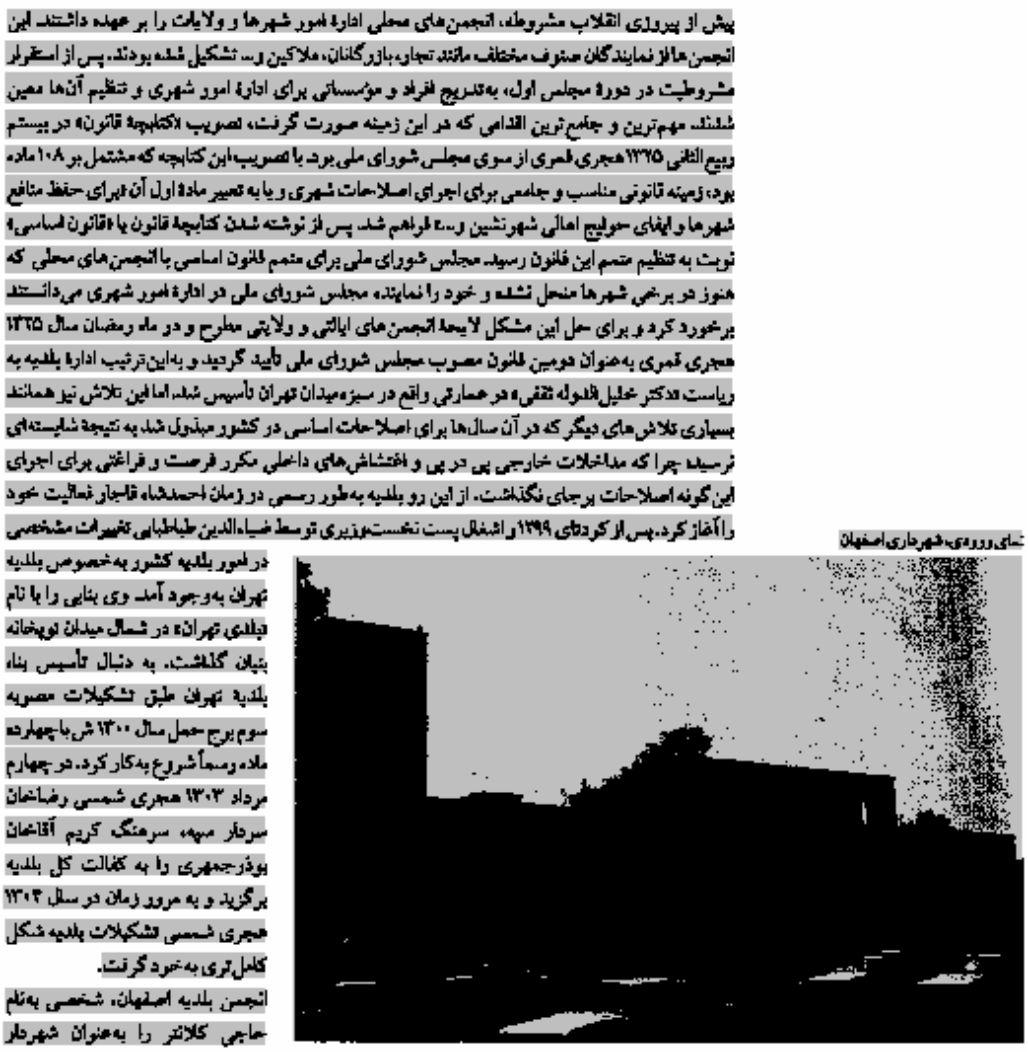


Fig. 12. The horizontal merging of connected components in region Fig. 11

At the second step, smoothing is applied to control amount of details in the projection profiles, as shown in (20) and (21), where s is the kernel size and m_y and m_x are the normalize value. Fig. 13(d) and 13(e) describes this example, which is result of smoothing for projection profiles in Fig. 13(b) and 13(c).

$$M_V^i = \{m_x \mid m_x = \frac{1}{s}\sum_{i=x-\frac{s}{2}}^{x+\frac{s}{2}} P_i, \quad 0 \le x < W, P^i \in P_V^i\} \tag{20}$$

$$M_H^i = \{m_y \mid m_y = \frac{1}{s}\sum_{i=y-\frac{s}{2}}^{y+\frac{s}{2}} P_i, \quad 0 \le y < H, P^i \in P_H^i\} \tag{21}$$

At third step, the smoothed projection profiles, M_V^i and M_H^i , are transformed to binary signals as described in (22) and (23). The binary signals of the smoothed projections profiles Fig. 13(f) and 13(g) are shown in Figure 13(d) and 13(e).

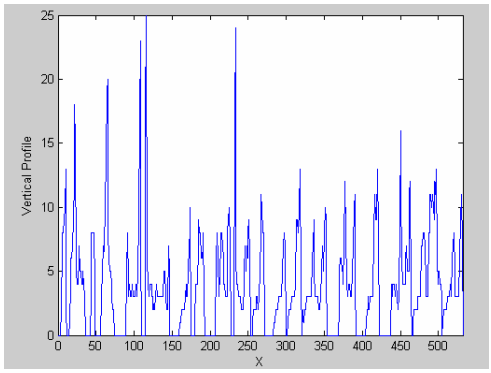
$$B_V^i = \begin{cases} 0 & m_x \le 0.01 \\ 1 & m_x > 0.01 \end{cases} \quad 0 \le x < W, m_x \in M_V^i \tag{22}$$

$$B_H^i = \begin{cases} 0 & m_y \le 0.1 \\ 1 & m_y > 0.1 \end{cases} \qquad (23)$$

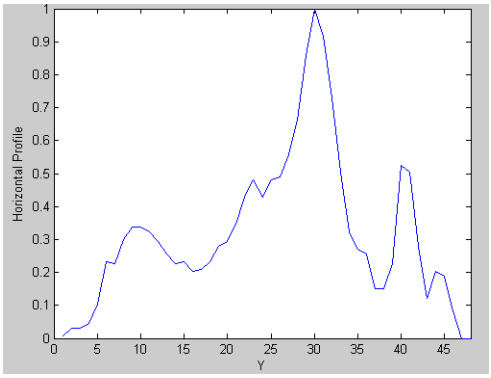
$$0 \le y < H, m_y \in M_H^i$$

If B_H^i signal have exactly one positive pulse and B_V^i signal have at least two positive pulse then GTL^i will be classified to text line (TL), otherwise classified to non text line (NTL).

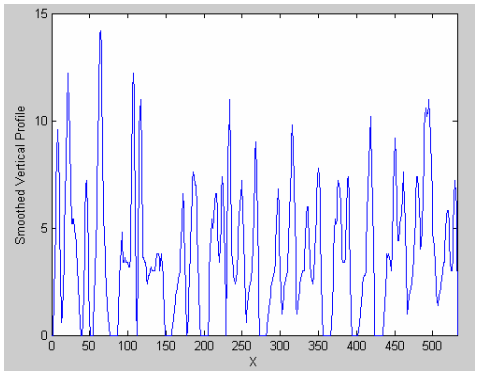
a) A GTL from Fig. 12



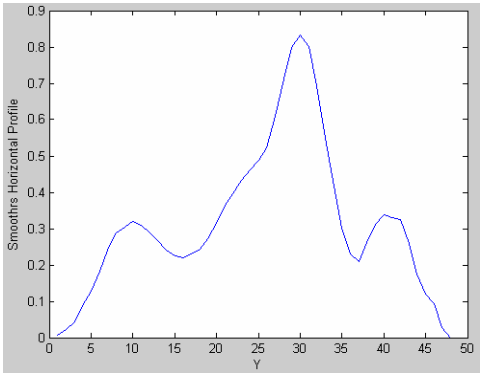
b) The vertical profile



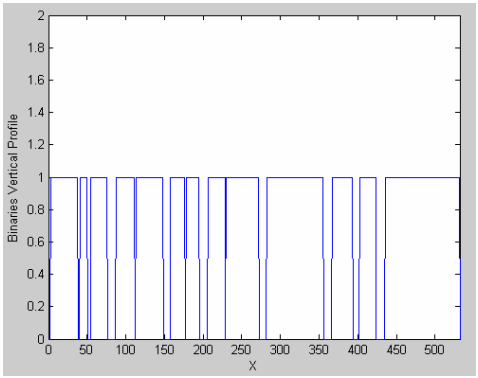
c) The horizontal profile



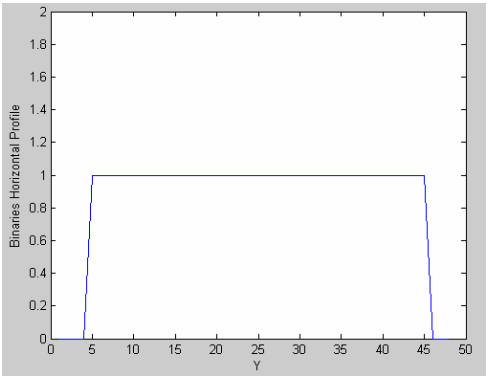
d) The smoothed vertical profile



e) The smoothed horizontal profile



f) The binaries vertical profile



g) The binaries horizontal profile

Fig. 13. The vertical and horizontal profile for a GTL

4.2.6. Vertical merging of TLs

This stage are described more detail in section 2.4. In this stage, TLs of region R_i are divided into block of text region by algorithm of Fig. 4. Fig. 14 is shown result of vertical merging on TLs of Fig. 12. Where the text regions are highlight in red color and non-text regions in yellow color.

4.2.7. NTLs classification

For classify non text lines (NTLS), first we can compute the number A(NTLi) of black pixels in an image region NTLi, which is equal to summation of the areas of connected components involved in the region. In order words,

$$A(NTLi) = \sum_{h}\sum_{w}\sum_{l} W(C_l)H(C_l) \qquad (24)$$

An image region NTLi is called an image region if following relation is satisfied.

$$\frac{A(NTLi)}{W(NTLi)H(NTLi)} \geq 0.4 \qquad (25)$$

Then all of NTLs which satisfied in relation (25) are classified to image region and the remaining NTLs regions are classified to table and drawing regions.

Fig. 14. The result of Vertical merging

5. Experimental Results and Performance Evaluation

The proposal Persian page segmentation has been tested on more than 50 images which were scanned from Persian magazines, all the images are scanned at 300dpi and each document image has a resolution of about 2400×2900 pixels. Fig. 15 shows some of our experimental results, where the text regions are highlight in red colour and image regions in yellow colour. These test images contain text and image.

The performance in quantitatively evaluated as shown in table. Base on 50 document images.

Table 1
Performance evaluation

	Correct	Failed	Correct rate (%)
Region location	245	8	96.8
Text	425	5	98.2
Image	48	3	95.5
Table/Drawing	21	2	97.5

The performance of the proposed Persian page segmentation approach is evaluated and compared with two other well-known text extraction techniques, namely Jain method [19], and Pietikainen method [20]. A set of 54 real-life complex document images was employed for experiments on performance evaluation of text extraction. These test images include a variety of book covers, book and magazine pages, advertisements, and other real-life documents at the scanning resolution of 200 dpi to 300 dpi. These images are comprised of textual objects in various colors or illuminations, font styles and sizes, including sparse and dense textual regions, adjoined or overlapped with pictorial, watermarked, textured, shaded, or uneven illuminated objects and background regions.

For the quantitative evaluation of page segmentation performance, three measures, the text rate, the image rate, and the table/drawing rate, which are commonly used for evaluating performance in page segmentation, are adopted. They are respectively defined as,

$$\text{Text rate} = \frac{\text{Number of correctly extracted text}}{\text{Total extracted of region}}$$

$$\text{Image rate} = \frac{\text{Number of correctly extracted Image}}{\text{Total extracted of region}}$$

$$\text{Table/Drawing} = \frac{\text{Number of correctly extracted Table/Drawing}}{\text{Total extracted of region}}$$

We compute the text, image, and Table/Drawing rates for page segmentation results of test images in this study by manually counting the number of actual regions of the document image, total extracted text-like regions, image regions and the correctly extracted other regions, respectively. The experiments of quantitative evaluation were performed on our test database of fifty Persian document images. From the text extraction viewpoint, the text rate reveals the percentage of correctly extracted characters within each processed document image, while the two other rates represent the percentage of non text correctly extracted.

Table 2 depicts the results of quantitative evaluation of Jain and Pietikainen methods and the proposed approach. By observing Table 1, we can see that the proposed approach provides better page segmentation performance as compared to that of Jain and Pietikainen s methods.

Table 2
Experimental data of Jain Pietikainen methods and our proposed approach

Method	Text rate	Image rate	Table/Drawing rate
Jain	85%	90%	92.5%
Pietikainen	79.5%	82%	81.5%
our proposed approach	98.2%	95.5%	97.5%

6. Conclusion

We have introduced a hybrid page segmentation method that segments the Persian document into homogeneous regions. The proposal method was experiment with Persian magazines documents. Experiments results have shown that more accurate and speed results. This method focuses on the Persian page segmentation, which can be extended for other scripts such as English scripts. Also this work can be extended for special work such as license plate recognition, postal service, noisy documents and etc.

References

[1] A. Dawoud and M. Kamel, Iterative multi-model sub-image binarization for handwritten character segmentation, IEEE Transaction Image Processing 13 (9) (2004) 1223-1230.

[2] X. Ye, M. Cheriet, C. Y. Suen, Stroke-model-based character extraction from gray-level document images, IEEE Transaction Image Processing 10 (8) (2001) 1152-1161.

Fig. 15. Result of Persian page segmentation on a sample image

[3] A. Amin and S. Wu, A robust system for thresholding and skew detection in mixed text/graphics documents, Int'l. J. Image and Graph. 5 (2) (2005) 247-265.

[4] J. Ha, R. Haralick, and I. Phillips, Recursive X-Y Cut Using Bounding Boxes of Connected Components, Proc. Third Int'l Conf. Document Analysis and Recognition (1995) 952-955.

[5] J. Ha, R. Haralick, and I. Phillips, Document Page Decomposition by the Bounding-Box Projection Technique, Proc. Third Int'l Conf. Document Analysis and Recognition (1995) 1119-1122.

[6] Y. Xiaoa, H. Yana,Text region extraction in a document image based on the Delaunay tessellation", Pattern Recognition, pp. 799-809, 2003.

[7] J. Xi , Jianming Hu, Lide Wu, Page segmentation of Chinese newspapers , Pattern Recognition (2002) 2695-2704.

[8] A. Jain and Y. Zhong, Page Segmentation Using Texture Analysis, Pattern Recognition 20 (1996) 743-770.

[9] A. Jain and S. Bhattacharjee, Text Segmentation Using GaborFilters for Automatic Document Processing ,Machine Vision and Applications 5 (1992) 169-184.

[10] M.Acharyya, M.K.Kundu, Document Image Segmentation Using Wavelet Scale-Space Features, IEEE Transaction on circuits and systems for video technology 12 (12) (2002).

[11] R.C.Gonzalez, R.E.Woods, Digital Image Processing, Second Edition, 2002, by Prentice-Hall.
[12] Y. M. Y. Hasan, and L. J. Karam, Morphological text extraction from images, IEEE Transaction Image Processing, 9 (11) (2000) 1978-1983.
[13] A. Jain and B. Yu, Document Representation and Its Application to Page Decomposition, IEEE Trans. Pattern Analysis and Machine Intelligence 20 (1998) 294-308.
[14] L.O'Gorman, and R.Kasturi, Document Image Analysis, IEEE computer Society Press, Los Alamitos, California, 1995.
[15] W.Postl,, Detection of linear oblique structures and skew scan in digitized documents, Proc. 8th Int. Conf. Pattern Recognition, Paros (1986) 687-689.
[16] S.N.Srihari , V.Govindraju, Analysis of textual image using the Hough transform, Machine Vision Application 2 (1989) 141-153.
[17] A.Hashizume, P.-S.Yeh, and A.Rosenfeld, A method of Detecting the orientation of Aligned Components, Pattern Recognition Letters, 4 (2) (1986) 125-132.
[18] B. Gatos, N. Papamarkos, and C. Chamzas, Skew Detection and Text Line Position Determination in Digitized Documents, Pattern Recognition 30 (1997) 1505-1519.
[19] A. K. Jain, and B. Yu, Automatic text location in images and video frames, Pattern Recognition 31 (12) (1998) 2055-2076.
[20] M. Pietikinen and O. Okun, Edge-based method for text detection from complex document images, in Proc. 6th Int'l Conf. Document Analysis and Recognition, 2001, pp. 286-291.