A Hybrid Approach to Localize Farsi Text in Natural Scene Images

Maryam Darab a, Mohammad Rahmatia*

a Department of Computer Engineering, Amirkabir University, Tehran, Iran

Abstract

Text in scene images can provide useful and vital information for content-based image analysis. Therefore, localization of text in images is an important task. In this paper, we present a hybrid approach to localize Farsi text in natural scene images. Complex background, variations of text font, size and line orientation and non-uniform illumination are the problems of this method. The Language of text localization in the past works is almost limited to English or Chinese. In this paper we consider Farsi/Arabic language for text localization. Due to the specific features of this language challenges of text localization are numerous. In this paper, in the first step a new color based method is proposed for extracting candidate regions, then the texts in natural scene images are detected by combining edge and color features. Variation due to text size and orientation, are resolved by a new pyramid of images. The candidate texts are verified by combination of two features, wavelet histogram and histogram of oriented gradient. Experimental results using our large dataset have demonstrated that the proposed method is effective and promising.

© 2012 The Authors. Published by Elsevier B.V.

Keywords: Natural scene images, Text localization, Text verification.

1. Introduction

Text detection in natural scene image has attracted researcher’s attention for many years. Texts provide more intuitive information and they are closely related to the content of image. So it is natural and convenient to
analyze semantic of image based on the information of texts. Text recognition include many applications such as: license plate reading, sign detection and translation, helping tourists and blind persons to understanding environment, drawing attention of a driver, content-based image search and so on.

As shown in fig. 1, text understanding system consists of four stages: text detection, text localization, text extraction and text recognition. Among these stages, text detection and localization are critical to the overall system performance [1].

In the last decade, many significant achievements have been made by researchers in the field of text localization (as surveyed in [1], [2]). However, most of the previous work present methods for text localization in image or video, but accurate text localization in natural scene image is still a challenge due to imaging conditions (viewing angle, blur, lighting, Resolution, Aliasing) and considered scenes (variations of text font, size, color and alignment orientation, complex background, illumination changes). On the other hand, most of the previous works, consider the texts in English or Chinese. But detection of Farsi or Arabic text in natural scene image has attracted a little. In this paper, we propose a method for localization of Farsi/Arabic texts in natural scene images.

We focus on Farsi language which is a right to left script. Farsi letters are used in several languages, like Arabic, Urdu and Pashto, (more than half of billion people). In addition, most of Muslims (almost ¼ of the people on Earth) can read Arabic because it is the language of Quran, the holy book of Muslims. Farsi language has four letters more than Arabic. Farsi characters can have more than one shape according to their position inside the word: initial, middle, final, or standalone. No upper or lower case exists in Farsi. Also, in Farsi there are characters with different extension in various fonts.

Natural scene text is textual part of still images or video frames of a scene with no a priori knowledge of environment, lighting, objects supporting text, acquisition parameters and text itself. It could easily be viewed as text in real-world conditions without any constraints or assumptions [4].

The existing methods of text localization usually fall into two categories: connected component (CC)-based and region-based method. Region-based method hold the assumption that backgrounds are much smooth than text regions. So it is possible to classify text region and non-text region according to edge or texture intensity. These methods can classify as heuristic [5-10] and machine learning based methods [11-19]. CC-based method assumes some constraints of text regions, such as uniform colors, certain sizes, and spatial alignments, are satisfied. The main problem of this kind of method is that it is not universal for all images. Because color, size and shape of the text can vary greatly from image to image [20-23].

Our method, according to Fig. 2, consists of two steps. In first step, candidate texts are located. In second step, candidate texts are verified and classified in text and non text class. We proposed two methods, edge-
based and color-based in the first step, to locate candidate texts. It means that we use combination of two categories, region-based and CC-based, for localization candidate texts. Then, we use two features, wavelet coefficient histogram and HOG for classification text and non text, in second step. The remaining of this paper is organized as follows; the candidate text location algorithm is presented in Section 2. The Text Verification algorithm is presented in Section 3 and our experimental results are presented in Section 4. Finally, we will draw our conclusions in Section 5.

2. Candidate Text Location

For locating candidate texts, we propose two methods, the first method is an edge-based method and the second method is based on color. In the following we describe these two methods.

2.1. Edge-based Method

Normally, intensity of an image is a major source of information for text detection; however it is sensitive to lighting variations. On the other hand, the gradient of the intensity (edge) is less sensitive to lighting changes [4]. Therefore, we use ‘edge’ as the first feature in the coarse detection phase. Edge-based method that used in this paper is based on Chen’s method [1]. However, modifications are made to this method in order to adaptive it to natural scene images and Farsi text. These changes include adaptive setting threshold in sobel filter and using image pyramid with a transformation matrix. In the following we describe two task of edge-based method, Detection of text blocks and Text line localization in candidate text regions.

Detection of Text Block.

The first part of the text localization procedure consists of detecting text blocks characterized by short horizontal and vertical edges connected to each other. The vertical and horizontal edge maps are first computed from the directional second derivative zeros produced by a Sobel filter. Then, based on the type of edges, different dilation operators are used so that vertical edges extend in horizontal direction while horizontal edges extend in vertical direction [11].

The first modification of edge-based method is adaptive threshold in sobel filter. The main source of failure to achieve appropriate detection rate and low false alarm rate is reliance on a fixed threshold value to segment text from non-text because existing methods generally assume that text has a high contrast over the background. Determining a proper threshold value for low contrast natural images, however, is not easy as compared to high contrast images. It is also observed that setting a low threshold in order to capture low contrast text will result in merger of multiple text lines, leading to more false positives. On the other hand, if the threshold is set too high, then low contrast text may be missing. Hence, at least two thresholds such as one for low contrast images and another for high contrast images need to be used to improve the performance of the detection methods. In order to fix two thresholds, classification of low contrast and high contrast images is necessary. Therefore, we use Otsu method for classification of high contrast and low contrast images. If result of applying Otsu method on image, is less than 0.5, the image is low contrast, otherwise is high contrast. After classification, edge Threshold in Sobel filter for high contrast and low contrast images is 0.1 and for low contrast images is 0.05.

Fig. 3 (b) and (c) displays the vertical and horizontal edges resulting of this process for the natural image showed in Fig. 3(a). The vertical and horizontal edge dilation results are shown in Fig. 3(d) and 3(e). Due to the connections between character strokes, vertical edges contained in text-like regions should be connected with some horizontal edges, and vice versa, we consider only the regions that are covered by both the vertical and horizontal edge dilation results as candidate text regions. Thus, Fig. 3 (f) illustrates the result of this step.
Text Line Localization in Candidate Text Regions.

The second part aims at extracting individual text lines from candidate text blocks. In order to normalize text sizes, we need to extract individual text lines from paragraphs in candidate text regions. This task can be performed by detecting the top and bottom baselines of horizontally aligned text strings. Baseline detection also has two additional purposes. Firstly, it will eliminate false alarms, such as slant stripes, which do not contain any well-defined baselines. Secondly, it will refine the location of text strings in candidate regions that contain text connected with some background objects. This baseline detection is performed by three splitting algorithms,

- Varying length text lines splitting algorithm. This algorithm is used to split text lines from connected background regions or split two text lines of very different lengths
- Equal length text lines splitting algorithm aims at splitting two text lines of similar lengths.
- Baseline refinement algorithm is employed when a text line cannot be split any more. Its aim is to extract the top and bottom baselines more accurately.

These splitting algorithm are described in Ref. [11]. Fig. 4(a) illustrates the result of applying this text line localization step in Fig. 3(f). Typical characteristics of text strings are then employed to select the resulting regions and the final candidate text line should satisfy the following constraints: it contains between 75 and 9000 pixels; the horizontal–vertical aspect ratio is more than 1.2; the height of the region is between 8 and 35. Fig. 4(b) shows the rectangle boundaries of the candidate text lines.

Fig. 2. Algorithm proposed for text localization

Pyramid of Images.

One of the challenges in natural scene images is various font sizes in a wide range. On the other hand, orientation of text is assumed horizontal in many papers. However, this assumption is very limited in natural scene images. Fig. 5 shows some natural scene images with various scale and orientation. As the second modification in edge-based method, to facilitate the detection of various text sizes and orientations, a pyramid of images is generated from the original image by applying a transformation matrix at each level. The number of levels was empirically determined to be seven. Transformation matrix defined as:
Where $S_x$, $S_y$ are scale parameters and $Sh_x$, $Sh_y$ are shearing parameters, along the x and y axis respectively. These parameters have following values in different layers of pyramid:

1- $Sh_y = 0, Sh_x = 0, S_x = 1/2, S_y = 1$
2- $Sh_y = 0, Sh_x = 0, S_x = 1/2, S_y = 1/2$
3- $Sh_y = 0, Sh_x = 0, S_x = 1/4, S_y = 1/4$
4- $Sh_y = 0.3, Sh_x = 0, S_x = 1, S_y = 1$
5- $Sh_y = -0.3, Sh_x = 0, S_x = 1, S_y = 1$
6- $Sh_y = 0.3, Sh_x = 0, S_x = 1/2, S_y = 1/2$
7- $Sh_y = -0.3, Sh_x = 0, S_x = 1/2, S_y = 1/2$

Therefore, in first level we have original image. In second and third layers, the image is obtained by reducing the image one level below by half in both dimensions. In fourth and fifth layers, the original image shears as 0.3 and -0.3 along the y axis, respectively. Sixth and seventh layers are similar to fourth and fifth, but image dimension is reduced by half.

For each pyramid level, we perform text candidate detection and verification (next step) with the same parameters. Then, these results from the different levels are grouped together. The main problem occurs when the same text line is detected together in two or three pyramid levels. In this case, we have to select the best one. To accomplish this, we perform pyramid composition based on the SVM output score. The result with the higher SVM output score is selected. This method is reasonable, because the highest SVM output score means the best localization of text [15].

![Candidate text region extraction. (a) original image; (b) vertical edges detected in image (a); (c) horizontal edges detected in image (a); (d) dilation result of vertical edges using 8×1 vertical operator; (e) dilation result of horizontal edges using 8×16 horizontal operator; (f) candidate text regions.](image-url)
Although edge-based method reported encouraging performance, it still needs further improvements. Our approach fails on some hard-to-segment texts as shown in Fig. 6, because of poor text edges. On the other, it fails when there are characters with different extension in various fonts, because of Farsi or Arabic specification. One possible solution is to take into account color information. Therefore, we propose a new color-based method, to extract text in hard-to-segment images. Next section describes this method.

2.2. Color-based Method

Because of problem in edge based method, for specific feature of Farsi language and lighting condition, we should use the color-based method for detection of candidate text. By assuming that either text or its background is uniform in color, we can locate candidate text by grouping text pixels through image segmentation and region layout analysis.

In image segmentation, a K-means algorithm is used to group pixels of similar color into the same cluster in L*a*b* color space. In this process, the key issue is how to decide the color quantization number ($N_{Quan}$). A method based on color variance analysis of the whole image is proposed in [16] to calculate $N_{Quan}$.
Supposing that there are $M$, $w_1 \times w_2$ windows in an image and each window contains $n$ pixels, we define $S_m$ as the color coarseness of a window, which represents the color variance of the window. Generally, the larger the color variance is, the more the color number should be. We can use the average color coarseness of the whole image $S_{avg}$ to calculate $N_{Quan}$. $S_m$ and $S_{avg}$ can be calculated from the following equations:

$$S_m = \left( \frac{1}{n} \sum_{i=0}^{n-1} \| \bar{x}_i - \bar{x}_{mean}^{(m)} \| \right)^{1/2}, \quad S_{avg} = \frac{1}{M} \sum_{m=0}^{M-1} S_m$$

where $\bar{x}_i$ is color values of pixel $i$ in $w_1 \times w_2$ and $n$ is number of color bands. In our experiments, we set $w_1$ and $w_2$ to be $1/10$–$1/20$ of the width and height of the image. $\bar{x}_{mean}^{(m)}$ is the average color values in a window. $\| \|$ is Euclidian distance. The larger $S_{avg}$ is, the larger $N_{Quan}$ should be. $N_{Quan}$ can be calculated by a linear function of $S_{avg}$ as

$$N_{Quan} = \alpha S_{avg} + 1$$

where $\alpha$ is a coefficient which can be set as 0.5 empirically. Once $N_{Quan}$ is decided, we can use the K-means algorithm to cluster pixels into $N_{Quan}$ color clusters. After a region growing operation, image pixels of the same color label and spatial connection are segmented into the same region. In Fig. 7a, the number of color for segmentation, according to equation 2, is 4. In Fig. 7b pixels of original image (Fig. 7a) labeled by the result of clustering of colors. Fig. 7c-f shows the color segmentation of original image.

After the image segmentation process, a spatial layouts analysis procedure is developed to obtain candidate text lines. The procedure includes:

1- Morphological ‘dilation’ operation: dilation operator is used to connect text pixels into text regions.
2- Finding seed pixels: pixel $P$ will be a seed pixel if the percentage of candidate pixels in its neighborhood is larger than the threshold. Eliminate pixels that don’t have this condition.
3- Refinement of candidate text lines: A horizontal projection profile is defined as the sums of the candidate pixels over rows. To separate the two text lines, we need to find the ‘valley’ on the profile where the profile value is smaller than a threshold $T_P$ and then segment the two text lines at the valley. $T_P$ is calculated as,

$$T_P = \frac{(Avg_{profile} + Min_{profile})}{2}.$$

Where $Min_{profile}$ and $Avg_{profile}$ are the minimum value and average value of the profile, respectively [17].

4- Heuristic constraints: the final candidate text line should satisfy the following constraints: it contains between 75 and 9000 pixels; the horizontal–vertical aspect ratio is more than 1.2; the height of the region is between 8 and 35.
Using the above procedure, most of the character components are merged into character regions. These regions need to be further connected into text lines by the following procedure:

1- Iteratively search region pairs that can meet the following conditions and connect them:
   — Regions have the same color label;
   — Regions are horizontally “adjacent” (maximum of horizontal and vertical distance of two regions is small enough, that is this distance must be less than 0.3*height of the larger region);
2- If there are no region pairs meeting the above conditions, procedure exists.

After the region spatial layout analysis, candidate text lines are located. The region layout analysis procedure is Adaptive to text font-size since all of the thresholds in the procedure are calculated in terms of the sizes of the regions themselves. However, in these located candidates, there will be many false alarms. Therefore, we must use a supervised classification procedure to identify true text. Fig. 8 shows the result from each step of color-based method on Fig. 8c-f. According to equation 2, the number of color for segmentation is four. Fig. 9 illustrates the results of this method.

In color based method, text lines with its color similar to the background, is unable to detect. Hence, to improve the performance, we use two methods, edge and color together for candidate text localization.

3. Text Verification

The candidate text location procedure described in the previous section is rather empirical and may therefore produce false alarms (i.e. non text regions). To remove these false alarms, we must use a supervised classification procedure to identify true text. To accomplish this, we chose to use an SVM rather than an NN, because it has been demonstrated that the SVM is better than the NN in text localization [15].

To classify text and non text regions, many features have been used in other work. These features consist of: gray-scale spatial derivatives, distance map, constant gradient variance, DCT coefficients [11], normalized gray intensity [18], wavelet coefficient histogram, color variance [16], wavelet moments, wavelet co-occurrence, scan line features [17] and others.
Fig. 8. Text location using color based method. (a) Dilation operation and determine seed pixel; (b) Refinement candidate text lines; (c) Heuristic constraints; (d) Region spatial layout analysis; (e) Corner constraint

Text can be considered as a special texture made up of positive and negative sharp signals [16]. On the other hand, Histogram of Oriented Gradient (HOG) and its variations have been widely used in computer vision [24] and optical character recognition (OCR) fields, due to their strong ability to describe the strength and regularity of object contours [16]. Based on these observations, wavelet coefficient histogram features and HOG are extracted to represent text pattern.

Wavelet coefficients can be used to effectively identify sharp signals in that filters in 2D wavelet transformation can properly capture the both signal variation and their orientations. Wavelet coefficients in wavelet LH and HL bands [25] are used to calculate the histogram \( H_i, i = 0, \ldots, 15 \) wavelet coefficients of all pixels. These coefficients are quantized into 16 levels:

\[
C_q = C.16/(C_{max} - C_{min})
\]
where $C$ is the wavelet coefficient of a pixel, $C_{\text{max}}$ and $C_{\text{min}}$ are the maximum and minimum values of wavelet coefficients in corresponding band, respectively. The wavelet histogram can then be computed on the quantized coefficients. The value of $H_i$ is the percentage of the pixels whose quantized coefficient is equal to $i$.

In the implementation of HOG, each pixel’s gradient vector, calculated by Sobel operator, can be decomposed into 4-orientation as shown in Fig. 10c or 8-direction as shown in Fig. 10d based on the parallelogram law. Within a specific local region, the feature value of each HOG bin is calculated by accumulating all corresponding bin values on all pixels. To avoid sensitivity to illumination, all HOG feature values are normalized by dividing intensity standard deviation (STD) value of the sampled window.

To improve the classification performance, we fuse the HOG and wavelet coefficient histogram features together. Because the SVM output score is the sum of weighted kernel distances between the test sample and support vectors, it is reasonable to fuse these two features in the following way:

$$ F = p_1 \cdot f_1 + p_2 \cdot f_2 $$

where $F$ is the final SVM score, $p_1$ and $p_2$ are the prediction accuracies, and $f_1$ and $f_2$ are the SVM scores of the two features, respectively.
4. Experiment Results

Since language of text is Farsi or Arabic, we collected a dataset of 800 images captured from natural scene for our experiments with canon camera. We believe that our dataset represents most of the text in real world. 500 images are selected for building the training set and 300 images are used for performance evaluation. The test set consists of a variety of situations such as text in different font-size, color, light text on dark background, text with textured background, and text of poor quality. Readers can obtain the dataset through the authors.

Given the ground truth and detection result by the algorithm, we can calculate the recall rate, precision rate and F1 by

\[ \text{Recall} = \frac{\text{Number of correctly located text lines}}{\text{Number of text lines}} \]

\[ \text{Precision} = \frac{\text{Number of correctly located text lines}}{\text{Number of located text lines}} \]

\[ F_1 = \frac{2(\text{Recall} \times \text{Precision})}{\text{Recall} + \text{Precision}} \]

Table 1 shows the result of candidate text localization with edge, color and combination of these two methods. It can be seen that the combination of edge and color methods, can greatly improve the recall rate. However, precision rate don’t change significantly. Since the goal of candidate text localization step is to extract potential text blocks in images with a very low missing rate and a reasonable false alarm rate, we use the combination of edge and color methods, because of high recall rate.

The results given in Table 2 show that the hybrid of two features, HOG and Wavelet histogram, gives higher precision and recall rates than the individual techniques. According to experiments, coefficients for HOG and Wavelet histogram in equation 5 are 2/3 and 1/3 respectively.

The proposed method showed robust performance on a majority of the test images. In Fig. 11, we illustrate some examples of the detected text lines. It can be seen from the results that most of the text is well detected despite of different font sizes and complex backgrounds. Fig. 11f shows an example of missing text line. This example shows the text line with its color similar to the background. It is missed because the color quantization procedure cannot correctly tell the text from the background. In real application this condition can be ignored since it is rare that text has similar color to its background. Fig. 11d, e shows a false alarm. In this image, since some tree branches appear quite like some characters, they pass both supervised classification. These false alarms can be eliminated by analyzing the lingual content of recognized text line in our future work.

The SVM was trained on a dataset consisting of 400 text and 1000 non-text labeled samples. Fig. 12 shows some of the training examples. As stated in [27], although positive samples are easy to be obtained, it is difficult to get representative negative samples since they may have various appearances. After a trained model is obtained, a “bootstrap” process is carried out to improve the performance of the classifier. That is to say, false alarms will be added into the training set for re-training. Our text detection has an overall recall of 86.5% and a precision of 80.8%.

Since our dataset is new and there isn’t work that localizes Farsi text in natural scene images, we couldn’t compare our method with other approaches.

Table 1. Comparison of edge and color based methods

<table>
<thead>
<tr>
<th>Method of localization</th>
<th>Recall rate (%)</th>
<th>Precision rate (%)</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge</td>
<td>77.5</td>
<td>33.2</td>
<td>46.5</td>
</tr>
<tr>
<td>Color</td>
<td>80.7</td>
<td>28.3</td>
<td>41.9</td>
</tr>
<tr>
<td>Edge + Color</td>
<td>86.5</td>
<td>29.4</td>
<td>43.9</td>
</tr>
</tbody>
</table>
Table 2. Comparison of features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Recall rate (%)</th>
<th>Precision rate (%)</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOG</td>
<td>76</td>
<td>76</td>
<td>76</td>
</tr>
<tr>
<td>Wavelet coefficient histogram</td>
<td>71.5</td>
<td>62.6</td>
<td>83.3</td>
</tr>
<tr>
<td>HOG + Wavelet coefficient histogram</td>
<td>83.5</td>
<td>80.8</td>
<td>86.5</td>
</tr>
</tbody>
</table>

Fig. 11. Experimental results. (a) Text with bad lighting condition; (b) text with various font sizes; (c) text with extension; (d) text with complex background; (e) false alarm; (f) missed text.

Fig. 12. Examples of the training data. (a) The text data; (b) Non-text data.
5. Conclusion

A new method for text localization in natural scene images is presented in this paper. Combining of edge and color is used to detect candidate texts. Then, wavelet coefficient histogram features and HOG are extracted to represent text pattern and SVM is used to classify text and non text components. Variation due to text size and orientation, are resolved by a new pyramid of images. Although, we design this text detection algorithm on Farsi text, it will be practical for English and Chinese. We collected a new complete dataset for Farsi text detection procedure.

Our next focus will be on introducing more features for representing text patterns and decreasing false alarms. On the other hand; we only provide a text detection method. Text should be clearly extracted from its background to obtain a good recognition result for the characters. Special technique should be investigated to segment the characters from their background before putting them into OCR software in the future.

References


