2012 10th IAPR International Workshop on Document Analysis Systems

Text Detection in Natural Scenes with Salient Region

Quan Meng
Institute of Artificial Intelligence and Robotics
Xi'an Jiaotong University
Xi'an, P.R.China
mengquann@stu.xjtu.edu.cn

Yonghong Song
Institute of Artificial Intelligence and Robotics
Xi'an Jiaotong University
Xi'an, P.R.China
songyh@mail.xjtu.edu.cn

***Abstract*—In this paper, we present a novel approach to detect text in natural scenes. This approach is a type of bionic method, which imitates how human beings detect text exactly and robustly. Practically, human beings follow two steps to detect text: the first step is to find salient regions in a scene and the second step is to determine whether these salient regions are text or not. Therefore, two similar steps namely salient regions computation and text localization are used in our method. In the step of salient regions computation, a set of salient features including multi-sacle contrast, modified center-surround histogram, color spatial distribution and similarity of stroke width are used to describe an image, following with computation of salient regions based on the combination of Conditional Random Fields model and above features. Because sole letter rarely appear, in the step of text localization, salient regions are segmented and the connected components are grouped into text strings based on their features such as spatial relationships, color difference and stroke width. As an elementary unit, the text string is refined by connected component analysis. We tested the effectiveness of our method on the ICDAR 2003 database. The experimental results show that the proposed method provides promising performance in comparison with existing methods.**

Keywords- text detection; salient regions; conditional random fields

I. INTRODUCTION

Text plays an important role in daily life due to its rich information. As a result, automatic text detection in natural scenes has many attractive applications, such as visual impairment assistance system, tourist assistance system and Unmanned Ground Vehicle (UGV) navigation in urban environments. However, locating text from a complex background is very difficult, because of the variations of scale, font, color, lighting and shadow [1] [2]. In recent years, many approaches have been proposed and they can be classified into two categories: texture-based methods and region-based methods.

Texture-based methods [3] [4] are based on the observation that text in images have distinct textural characteristics from the background. In these methods, the image is generally scanned in multi-scales, then texture analysis approaches, such as Discrete Cosine Transform (DCT), Fourier transform, distribution of Wavelet, spatial variance or Gabor filters are used to obtain texture information.

Region-based methods [5] [6] are mainly based on bottom-up approach. In this approach, pixels exhibiting certain similarities in color and intensity are first grouped, whereas non-text connected components (CCs) are filtered out from the candidate components using geometrical analysis.

Although, these existing methods have some positive results, they still have some disadvantages in terms of computational complexity and low accuracy. For example, even though texture-based methods reduce the dependency on some heuristic rules, they suffer from their computational complexity in the texture classification stage. On the other hand, region-based methods can identify texts at any scale, but it is very hard to segment text components accurately from a complex background.

In this paper, we propose a new approach that imitates how human beings detect text in natural scene. It is inspired by how human beings overcome above difficulties. Generally, text in natural scene is apparent to be seen by human eyes. The regions that catch human beings' attention are defined as salient regions. Let's give an example to interpret the detection process by human beings. Suppose there is a cup and wallet with the same size and background. If we are interested in the cup, the first object we pay attention to is the cup, and vice versa. Human beings are always paying attention to the objects which are bright and special, if they do not have a priority to find certain object. However, if people are interested in specific object such as text, they combine particular features of text and the common salient features, such as brightness, together subconsciously and use these features to identify salient regions and determine whether the salient regions are text or not.

Many approaches for visual attention have been proposed. *Laurent Itti* [8] proposed an approach which is based on the bottom-up computational framework. *Tie Liu* [9] applied three novel features (multi-scale contrast, center-surround histogram, and color spatial distribution) to describe a salient object. After that a conditional random field model is established to combine these features.

These visual attention approaches work well in salient object detection. However, what we are interested is text.

978-0-7695-4661-2/12 $26.00 © 2012 Crown Copyright
DOI 10.1109/DAS.2012.85

Therefore, we should propose some novel features of text. We ran the three features presented by *Tie Liu* [9] on the public database available in Ref. [10]. The results show that the features of multi-scale contrast and color spatial distribution work well in text, but the feature of center-surround histogram does not. This is because text has many holes, where the difference between center and surround is not so obvious as ordinary object.

In this paper, we propose two novel features, similarity of stroke width and modified center-surround histogram, which are coordinated with the features of multi-scale contrast and color spatial distribution. The first feature is the particular feature of text whereas other three features describe text regionally, locally and globally, respectively. Conditional Random Fields model is utilized to compute salient regions based these four features. The application of this model allows us to adopt not only region-based features but also texture-based features rather than a single type of feature. ICDAR 2003 Database is selected to analyze the performance of our method, which flow chart is illustrated in Fig. 1. The results show that the proposed approach gives good performance.

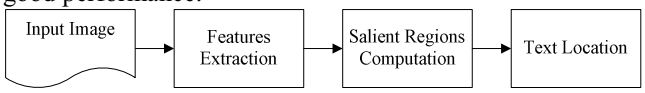


Figure 1. Flow chart of the proposed approach.

II. SALIENT REGION COMPUTATION

A. Salient Features

Similar to other vision problems, features play important roles in salient region computation. Here we combine the following four features to describe salient regions.

1) Multi-scale contrast

Generally, text that likes other object has obvious contrast feature. The size of text in natural scene is various, so we compute contrast at multiple:

$$f(x,I)=\sum_{l=1}^{L}\sum_{x'\in N(x)}\left\|I^l(x)-I^l(x')\right\|^2 \tag{1}$$

where L is the number of pyramid level, I^l is the l th-level image in the pyramid and x' is the neighbor of x in a 9×9 window. The result is normalized to [0,1] at the end.

2) Color spatial distribution

Text usually has similar color in natural scene. Furthermore, in order to capture people's attention, the difference of color between text and background is distinct in general. Therefore, spatial distribution of color is used to describe the global saliency of text.

Firstly, Gaussian Mixture Models (GMMs) are selected to represent all color in the image. Then the spatial variance of each color component is computed based on their horizontal and vertical variance in the spatial position. Last, the feature of color spatial distribution is computed using the following formula:

$$f(x,I)\propto\sum_{c}p(x|I_x)\cdot(1-V(c)) \tag{2}$$

where $p(x|I_x)$ is the probability that each pixel is assigned to a color component, which is defined as

$$p(c|I_x)=\frac{w_c N(I_x|u_c,\sum_c)}{\sum_c w_c N(I_x|u_c,\sum_c)} \tag{3}$$

and w_c u_c $\sum_c$ is the weight, the mean color and the covariance matrix of the c th component, respectively. $V(c)$ is the spatial variance of the c th component.

3) Modified Center-surround Histogram

Salient objects usually have distinct difference from their surroundings in intensity. Thus the χ^2 distance between histograms of intensity is used to describe the regional salient feature:

$$\chi^2(R,R_S)=\frac{1}{2}\sum\frac{(R^i-R_S^i)^2}{R^i+R_S^i} \tag{4}$$

where R is a rectangle that enclose the salient object and R_S is the surrounding with the same area as R.

Text is different from ordinary salient objects due to the existence of many holes, which have same intensity with background. Thus the difference between the center and the surrounding is not so obvious as ordinary objects. However, experiments show that the amount of edge pixels divided by the area of bounding box (r) is relatively fixed and ordinary objects don't have this characteristic. Therefore, we modify the center-surround feature using the following formula:

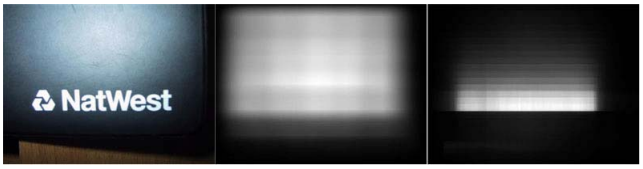
$$f(x,I)=g(r)\times f'(x,I) \tag{5}$$

where $f'(x,I)$ is the original center-surround feature, g is a Gaussian function.

To compute the intensity difference between the center and the surrounding, there are mainly two approaches to compare histograms, i.e. using the similarity measure L^1 norm or χ^2 difference, and using Earth Mover's distance (EMD). L^1 norm and χ^2 difference introduce quantization artifacts. The EMD can avoid quantization artifacts, but it is computationally expensive. So in this step, we combine the advantages of the above two approaches and adopt Gaussian function to smooth the histograms and prevent the quantization artifacts. Fig.2. shows one example of original and modified center-surround histogram. Obviously, modified center-surround histogram has a better result.

4) Similarity of Stroke Width

In order to get the similarity of stroke width feature, *Stroke Width Transform* (SWT) [11] is applied to compute the width of per pixel in the image, following with computation of similarity of stroke width based on the stroke width information. SWT first computes the edges of an image, then it follows the gradient direction of each edge



(a) (b) (c)

Figure 2. Modified center-surround histogram. (a)input image, (b)original center-surround histogram, (c)modified histogram.

pixel until another edge pixel is found. The distance between the two edge pixels is the stroke width of pixels along the segment of the two edge pixels when the direction of the two edge pixels is roughly opposite, unless the pixel already has a lower stroke width value. In the case of corner, the median SWT value (m) of each non-discarded ray is computed. If the pixels with SWT values below m are roughly similar and the amount of outliers (SWT values are too big) is little, the pixels whose SWT values is higher than m are assigned to m.

Edge detection is the fundamental procedure of SWT, but the result of Canny edge detector has many fractures, which influence the results seriously. Therefore, edge is repaired using morphology method after edge detection.

Using the SWT information, we group two neighboring pixels together, if they have similar stroke width. After that, outliers (too big or little) are eliminated in each CC. The following formula is used to get similarity of stroke width:

$$f_s(x, I) = \exp(\lambda \times sm) \tag{6}$$

where λ is the coefficient of exponential function, sm that evaluates the similarity of stroke width is the standard deviation of CC's stroke width which the pixel belongs to.

B. Compute salient regions

After the stage of salient features computation, we get four salient features. Conditonal Random Field (CRF) is used to compute salient regions based on these four features. The energy function of CRF is defined as:

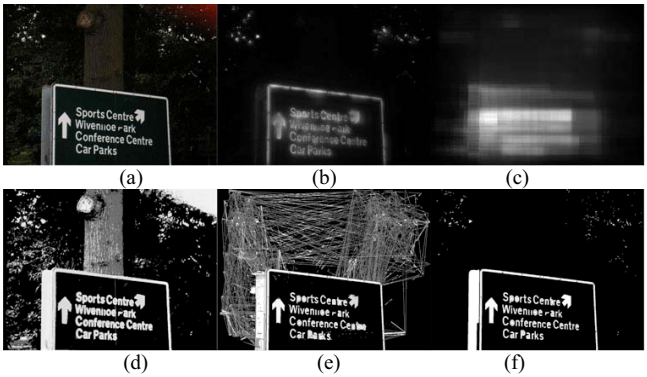
$$E(A|I) = \sum_{x} \sum_{k=1}^{K} \lambda_k F_k(a_x, I) + \sum_{x,x'} S(a_x, a_{x'}, I) \tag{7}$$

where K is number of features, λ_k is the weight of the kth feature, and x, x' are two adjacent pixels. $F_k(a_x, I)$ refers to the unary features transformed from the four salient features as follows:

$$F_k(a_x, I) = \begin{cases} f_k(x, I) & a_x = 0 \\ 1 - f_k(x, I) & a_x = 1 \end{cases} \tag{8}$$

where a_x indicates the label of the pixel.

$S(a_x, a_{x'}, I)$ is the pairwise feature which is based on the observation that if the color difference between two adjacent pixels is little, they are likely to be assigned with the same label. Fig.3 illustrates the feature maps and the labeling result of one example.



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

Figure 3. Salient features and salient results. (a) input image, (b) multi-scale contrast, (c) modified center-surround histogram, (d) color spatial distribution, (e) similarity of stroke width and (f) salient result.

III. TEXT LOCALIZATION

After the step of salient region detection, we get some candidate text regions. These candidate text regions may contain some non-text components, furthermore, these text components and non-text components are likely to have conglutination. Therefore, in this step we firstly separate text and non-text CCs from salient regions, and then filter out non-text CCs using CCs analysis method. Text usually appears in the form of text string in most natural scene. In this step, we adopt two methods to filter out non-text CCs from text CCs: 1) using single CC to train classifier and using CCs set to classify; 2) using CCs set to train classifier and to classify. We call these two methods as 'character train and string classify' and 'string train and string classify' respectively. These two methods share same framework:

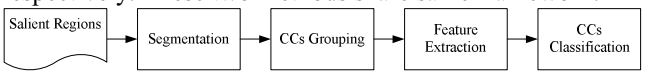


Figure 4. Flow chart of text location.

A. Character Train and String Classify

1) Salient Regions Segmentation

For the conglutination between text CCs and non-text CCs situations, we adopt Niblack's binarization algorithm[7] to segment them. We only compute salient regions, and set non-salient regions as background directly. As a result, we can not only improve computation speed, but also reduce the number of CCs. The window size of Niblack is set to half height of the CCs. An example of this stage is shown in Fig. 5.

2) Feature Extraction

After the step of salient regions segmentation, salient regions are separated into a set of CCs. Therefore fine text verification becomes a main issue of classification. To any classification problem, feature is the most important, thus we propose six features to describe characteristics of the CCs.

(a) (b) (c)

Figure 5. Example of salient region segmentation. (a) input image, (b) salient map computed from original image and (c) the result of segmentation.

a) Aspect Ratio Features is defined as the height divided by the width of the CC. The feature of character is in a fixed range, thus many too thin CCs are discarded.

b) Occupy Ratio describes how large area of the bounding box region that the CC's covers.

c) Contour Roughness is a feature used to filter noise with irregular shape but have strong texture response.

d) Compactness is defined as the area of CC divided by the square of CC's perimeter

e) Stroke Similarity is proposed to describe the property of stroke similarity, based on the observation that character stroke width is usually similar.

f) Stroke Size feature is defined as maximum of the mean/h and mean/w， and h and w is the height and width of the CC respectively.

3) CCs Grouping

It is well known that single letter does not usually appear in natural scene. Thus in order to increase the reliability of text verification and remove randomly scattered noise, we group CCs together and consider CCs set as the elementary analysis unit. Then CCs analysis method is used to verify whether CCs in the CC sets are text or not.

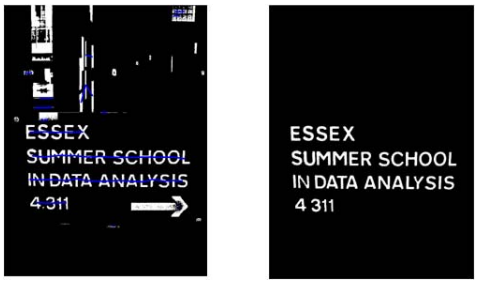
An important cue to CCs grouping is that text appears in a linear form in natural scene. What's more, text in a group is expected to have similar color, stroke width, letter height and width. Therefore, we use these criteria to merge separated CCs into groups.

4) Filter out non-text CCs

SVM is used to identify whether a CC in a CC set is text or not. The output of SVM is not forced to 0 or 1, but it is a value to describe the confidence of result. The following formula is used to do this.

$$\begin{cases} v_i > \theta & if\ n = 1 \\ v_i + mean(V) > 0 & other \end{cases} \quad (9)$$

where, V is the vector of value of SVM to a set of CC, v_i is the result of ith CC in this set, n is the amount of this set, θ is the threshold to judge whether a single CC is text or not. When the amount of a set is one, the criterion is very simple that whether the value bigger than the threshold (θ). When amount of a set is bigger than one, the effect of outliers can be eliminated by the mean of this set using this method.



(a) (b)

Figure 6. Examples of CC merging (with blue lines) and classification result.

At last, the text lines are broken into separate words, using the method proposed in [13].

B. String Train and String Classify

This method is similar to 'character train and string classify'. The two main differences are that CCs set is used to train SVM classification and CCs set's feature rather than CCs' feature is used to train and classify. Eight CCs set features are selected to describe a group of CCs. The first six features, which are derived from 'character train and string classify', compute the mean of CC features in a group. The last two features are based on the observation that the height of letter and spaces between letters are similar in a text string.

IV. EXPERIMENTS

In order to evaluate the performance of our method, we ran it on the public database of the ICDAR 2003 Robust Reading Competition, which contains 249 images with various size, color and font and complex background. For training SVM, we manually labeled 1741 text CCs and 5168 non-text CCs.

To evaluate the advantages and disadvantages of the two text location methods, we compared the two methods on the database. What's more, we also apply the method that uses single CC to train and to classify for comparison. Results are shown in Fig. 7.

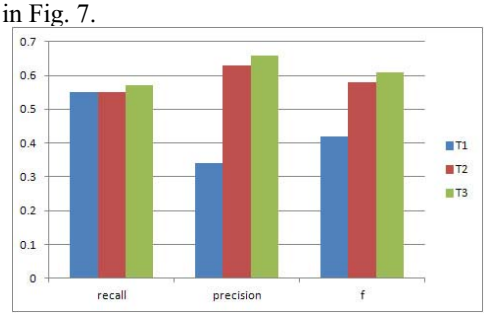


Figure 7. Performance of different text location. T1 indicates 'character train and character classify'. T2 indicates 'string train and string classify'. T3 indicates 'character train and string classify'.

The above results show that the performance of 'character train and string classify' and 'string train and string classify' are better than 'character train and character

classify'. These two methods can filter out many single CCs that are similar to text. The performance of 'string train and string classify' is lower than 'character train and string classify', because this method depends on the performance of grouping deeply. More importantly, 'character train and string classify' can get correct result even the grouping step is not correct.

To evaluate the proposed method, we adopted the performance evaluation criterion of the ICDAR 2005 competition. Table 1 shows that our method gives competitive result compared with existed method. However, the recall is not very good, because when text is too small or the difference between text and background is too little, the text is very salient. But fortunately, most text in real world is salient.

TABLE 1: TEXT DETECTION RESULT ON TEST IMAGES

	Precision	**Recall**	***f***
Hinnerk Becker	0.62	0.67	0.62
Our system	**0.66**	**0.57**	**0.61**
Alex Chen	0.60	0.60	0.58
Qiang Zhu	0.33	0.40	0.33
Jisoo Kim	0.22	0.28	0.22
Nobuo Ezaki	0.18	0.36	0.22

Figure 8 is some examples of our method. We can see that our method can effectively detect text in most case.

Figure 8. Example of text detection results.

V. CONCLUSION

In summary, a novel text detection algorithm by utilizing the salient information of natural scenes is proposed. We mainly imitate human beings' text detection in natural scenes. Firstly four salient features are used to describe a natural image, and then CRF is used to combine these features to get a salient map. In the fine text localization step, we view CC sets rather than single CC as the elementary unit. Experimental results show that the proposed method provides competitive performance in text detection.

ACKNOWLEDGMENT

This work was supported by the NSF of China (Grand no. 90920008).

REFERENCES

[1] K. Jung, K. I. Kim, and A. K. Jain, "Text information extraction in images and video: a survey," *Pattern Recognition* 37, no. 5, 977-997 (May 2004)

[2] Y. F. Pan, X. W. Hou, and C. L. Liu, "A robust system to detect and localize texts in natural scene images," in proc. *Eighth IAPR Workshop on Document Analysis Systems (DAS'08).* Pages 35-42.Nara. Japan. 2008.

[3] X. Chen, and A. Yuille, "Detecting and Reading Text in Natural Scene," *Computer Vision and Pattern Recognition (CVPR)*. Pp.366-373. 2004

[4] Q. Ye. Q Huang, W. Gao, D. Zhao, "Fast and robust text detection in images and video frames," *Image and Vision Computing* 23 (2005) 565-57

[5] J. Zhang and R. Kasturi, "Extraction of text objects in video documents: Recent progress," in proc, *Eighth IAPR Workshop on Document Analysis Systems (DAS'08)*. Pages 1-13, Nara, Japan, 2008

[6] A. Jain, and B. Yu, "Automatic Text Location in Images and Video Frames", *Pattern Recognition* 31 (12): 2005-2076 (1988)

[7] W. Niblack, "An Introduction to Digital Image Processing," *Strandberg Publishing Company*, Birkeroed. Denmark. 1985

[8] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans, on PAMI*.20 (11): 1254-1259. 1998.

[9] T. Liu, J. Sun, N. N. Zheng, X. Tang, and H. Y. Shum, "Learning to detect a salient object," *IEEE conference on Computer Vision and Pattern Recognition*,2007.

[10] http://algoval.essex.ac.uk/icdar/Datasets.html.

[11] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting Text in Natural Scenes with Stroke Width Transform," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE,2010, pp.2963-2970.

[12] N. Ezaki, M. Bulacu, and L. Schomaker, "Text detection from natural scene images: Towards a system for visually impaired persons," *Pattern Recognition*, vol. 2, pp.683-686, 2004.

[13] S. Karaoglu, B. Fernando, and A. Tremeau, "A Novel Algorithm for Text Detection and Localization in Natural Scene Images," *International Conference on Digital Image Computing*, 2010.

[14] Y. F. Pan. C. L. Liu ,X. W. Hou, "Fast Scene Text Localization by Learning-based Filtering and Verification," In proc. *17th ICIP* 2010.