International Journal of Engineering Research and General Science Volume 3, Issue 3, Part-2 , May-June, 2015
ISSN 2091-2730

Identification of Scene Text by Character Descriptor in Smart Mobile Devices

Devdas[1], Bhavana.S[2], Dr. Shubhangi D.C.[3]

Student, Department of computer science and engineering, VTU RO Kalaburagi, India[1]

Assistant professor, Department of computer science and engineering, VTU RO Kalaburagi, India[2]

Head of Department, Department of computer science and engineering, VTU RO Kalaburagi, India[3]

Email:Devd.bansode@gmail.com. Contact no: 8951781387

Abstract— Text data present in images and video contain useful information for automatic annotation, indexing, and structuring of images. Extraction of this information involves detection, localization, tracking, extraction, enhancement, and recognition of the text from a given image. However, variations of text due to differences in size, style, orientation, and alignment, as well as low image contrast and complex background make the problem of automatic text extraction extremely challenging. The main focus of this system is on two character recognition methods. In text detection, previously proposed algorithms are used to search for regions of text strings. Proposed system uses character descriptor which is effective to extract representative and discriminative text features for both recognition schemes. The local features descriptor HOG is compatible with all above key point detectors. Our method of scene text recognition from detected text regions is compatible with the application of mobile devices. A personal digital assistant (PDA) was chosen because it combines small-size, computational resources and low cost price. Three key technologies are necessary: text detection, optical character recognition and speech synthesis. The demo system gives us details of algorithm design and performance improvements of scene text extraction. It is able to detect text region of text strings from cluttered and recognize characters in the text regions.

Keywords— Scene text detection, scene text recognition, character descriptor, stroke configuration, text understanding, text retrieval mobile application.

1. INTRODUCTION

Text in the image contains useful information which helps to acquire the overall idea behind the image. Character extraction from image is important in many applications. It is a difficult task due to variations in character fonts, sizes, styles and text directions, and presence of complex backgrounds and variable light conditions. Several methods for text (or character) extraction from natural scenes have been proposed. If develop a method that extracts and recognizes those texts accurately in real time, then it can be applied to many important applications like document analysis, vehicle license plate extraction, text- based image indexing, etc. Camera-based applications on mobile phones are increasing rapidly. There can be valuable information in image. However, extraction of text from scene image is problematic due to factors such as variety of scale, orientation, font, style of character and complex background with multiple colors. Text Recognition in natural scene images is challenging than recognizing text from scans of printed pages, faxes and business cards. Modeling character structure is difficult due to high variability of geometry and appearance of characters natural image. To solve these problems text extraction is divided in two activities [9]: text detection and text recognition. Text detection localize region containing text characters [4]. Text recognition distinguishes different characters which are part of text word. We have presented two schemes text recognition process. First, a character recognizer to predict the category of a character in an image patch. Second, binary classifier which predicts existence of category. Two schemes of text recognition are compatible with applications related to scene text, which are text understanding and text retrieval. Text understanding acquires text information from natural scene to understand surrounding environment and objects, while text retrieval matches some stated user query against a set of free-text records. we design a discriminative character descriptor by combining several state-of-the-art feature detectors and descriptors[6].We model character structure at each character class by designing stroke configuration maps[5]. It involves 62 identity categories of text characters, including 10 digits [0-9] and 26 English letters in upper case [A-Z] and lower case [a-z]. An Android- based demo system is developed to show the effectiveness of our proposed method on scene text information extraction from nearby objects. Besides, previous work rarely presents the mobile implementation of scene text extraction, and we transplant our method into an Android-based platform. We propose an Android application that detects the text information within an image taken with a mobile phone camera, extracts it, recognizes it and translates it.

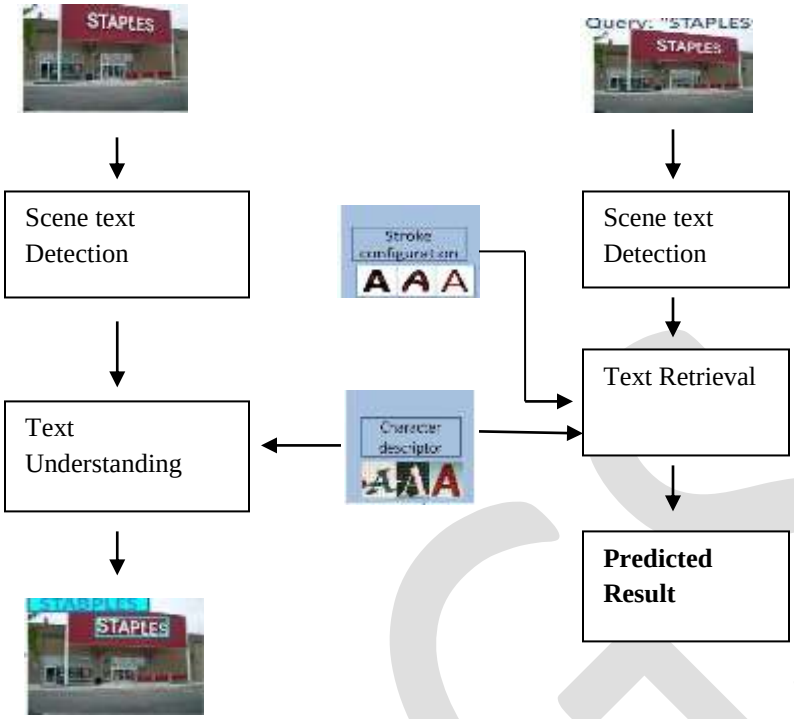

Fig. 1. The flowchart of our designed scene text extraction method

Objective of this system is the extraction of text from any image and then displaying its related information on the mobile screen. Main goal of this system is that if a person doesn't have or know any specific thing then he/she could get its information with the help of this android application**.**

In text extraction feature text is being extracted from the natural scene or an image. Here text extraction is done with the help of character description and stroke configuration [1]. Firstly the text will be detected, understood and then recognized. In searching process extracted text is being searched over net or in database. Here searching is done with the help of item ranking according to the item of interest. It basically derives meta data information about the item of interest by extending the user's given interest.

2. RELATED WORK

Our present a general review of previous work on scene text recognition respectively. Our observe that text characters from different categories are distinguished by boundary shape and skeleton structure, which plays an important role in designing character recognition algorithm. Extracting text from image is a difficult task. To perform this task various techniques have been implemented before. Cluster classification [1] is one of the techniques which have high accuracy in detecting text area and non-text area. There is new trend towards content based document image retrieval technique without going through OCR process [2].

There is another technique named as sliding window detection which has high accuracy of detecting text in natural scene. In Scale Invariant Feature Transform (SIFT), feature matching was adopted to recognize text characters in different languages, and a voting and geometric verification algorithm was presented to filter out false positive matches. SIFT reduces false positive rates by more than an order of magnitude relative to the best Haar wavelet based detector . Another source of information is found in the similarity and dissimilarity between pairs of characters. Weinman and Learned-Miller two characters which have nearly identical appearance have different labels [8].Text recognition system using above source of information have proved that two characters which have nearly identical appearance have different labels.

3. LAYOUT-BASED SCENE TEXT DETECTION

3.1 Layout Analysis of Color Decomposition

Test strings on signage boards consist of characters in uniform color and aligned arrangement. We can locate text information by extracting pixels with similar colors. A boundary clustering algorithm based on bigram color uniformity in our previous work [3]. Text boundaries on the border of text and its attachment surface are described by characteristic color-pairs, and we are able to extract text by distinguishing boundaries of characters and strings from those of background outliers based on color pairs. We then model color difference by a vector of color pair, which is obtained by cascading the RGB colors of text and attachment surfaces. Each boundary can be described by a color-pair, and we cluster the boundaries with similar color pairs into the sample layer. The boundaries of text characters are separated from those of background outliers.

3.2 Layout Analysis of Horizontal Alignment

In each color layer, we analyze geometrical properties of the boundaries to detect the existence of text characters. According to our observation, text information generally appears in text strings composed of several character members in similar sizes rather than single character, and text strings are normally in approximately horizontal alignment. This method involves following steps. Here we assume that length of signage and other text is enough to get benefit from repeatability of words while decoding. Here adjacent character grouping method is adopted from previous work [4]. For each connected component C we search for its siblings in similar size and vertical locations. When connected components C and C′ are grouped together as sibling components, their sibling sets will be updated according to their relative locations. When C is located on the left of C′, C′ will be added into the right-sibling set of C, which is simultaneously added into the left-sibling set of C′. For connected component C, if several siblings are obtained on its left and right, then we merge all these involved siblings into a region. This region contains a fragment of text string. To create sibling groups corresponding to complete text strings, we repeat above method to calculate all text string fragments in this color layer, and merge the string fragments with intersections.

4. STRUCTURE-BASED SCENE TEXT RECOGNITION

Our goal is to find the most likely word from this set of characters. We formulate this problem in an energy minimization framework, where the best energy solution represents the ground truth word we aim to find. The text retrieval schemes to verify whether a piece of text information exists in natural scene. In text retrieval, binary classifier distinguishes character class from other classes or background outliers. In text understanding character recognition is a multi-class classification problem. For each of the 62 character classes, we train a binary classifier to distinguish a character class from the other classes or non- text outliers. The specified character classes are defined as queried characters. In text retrieval, to better model character structure, we define stroke configuration for each character class based on specific partitions of character boundary and skeleton. In the text recognition technique as used the Optical Character Recognition (OCR) process is divided into following phases preprocessing, segmentation, feature extraction and classification. Fig 2. shows phases in classical OCR.

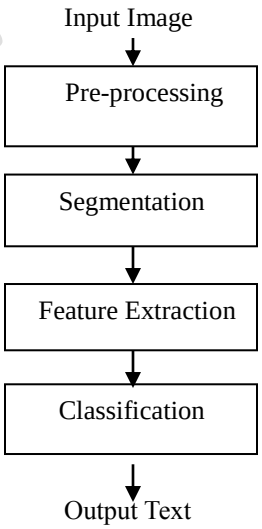

Fig .2.Optical Character Recognition

4.1 Character Descriptor

Four types of character descriptors are used to model character structure. Harris detector to extract key points from corners and junctions. MSER detector to extract key points from stroke components. Dense detector extracts key points uniformly. Random detector extracts present number of key points in a random pattern. By cascading BOW and GMM – based feature representations we get character descriptor as shown in figure 2 below. In GMM model the numbers and locations of key points from each patch should be identical. Therefore, it is only applied to the key points from DD and RD.

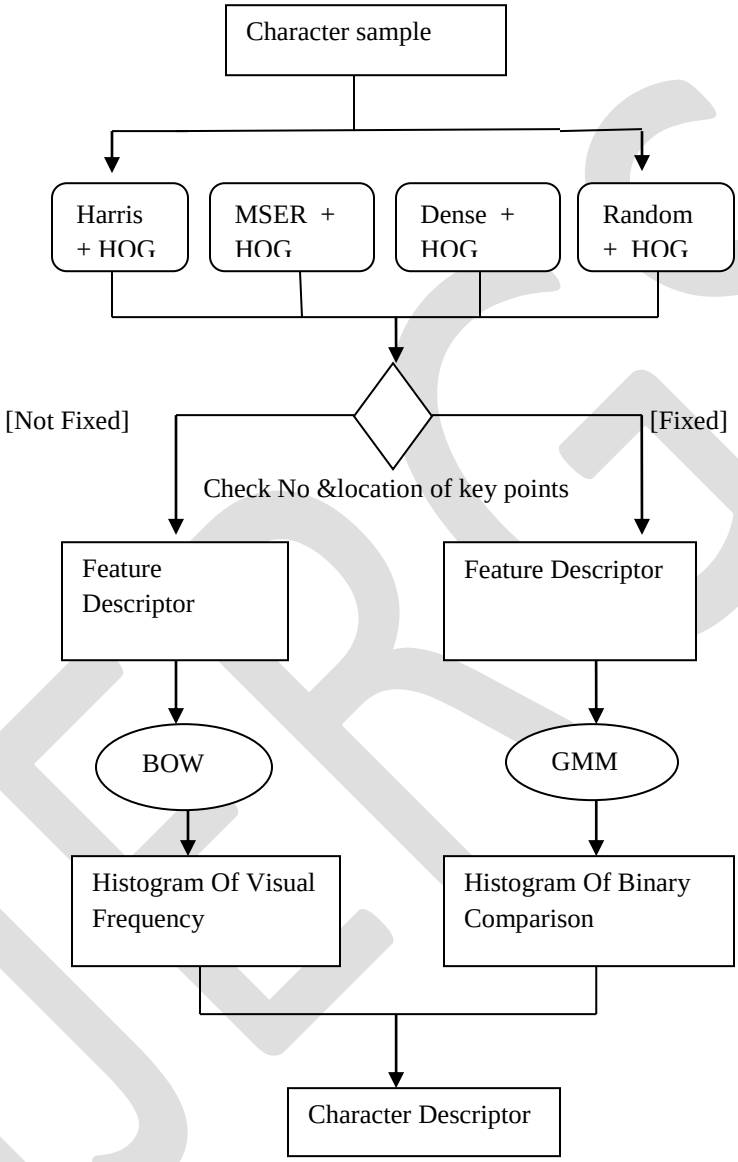

Fig.3. Flowchart of our proposed character descriptor.

Four feature detectors are able to cover almost all representative key points related to the character structure. At each of the extracted key points, the HOG(Histogram of Oriented Gradients) feature is calculated as an observed feature vector x in feature space. Each character patch is normalized into size 128 × 128, containing a complete character. In the process of feature quantization, the Bag-of-Words Model and Gaussian Mixture Model are employed to aggregate the extracted features. BOW model represent the frequency of word occurrence, but not their position. SIFT and SURE are not employed in our method because their performance on character recognition is low. Every character patch is normalized into size 128 × 128 containing complete character. In both models, character patch is mapped into characteristic histogram as feature representation.

4.1.1 BOW Model

The BOW representation is computationally efficient and resistant to intra-class variations. At first, k-means clustering is performed on HOG features extracted from training patches to build a vocabulary of visual words. Then feature coding and pooling are performed to map all HOG features from a character patch into a histogram of visual words. We adopt soft-assignment coding and average pooling schemes in the experiments. More other coding/pooling schemes will be tested in our future work.

At a character patch, the four detectors are applied to extract their respective key points, and then their corresponding HOG features are mapped into the respective vocabularies, obtaining four frequency histograms of visual words. Each histogram has 256 dimensions. Then we cascade the four histograms into BOW-based feature representation in 256 × 4 = 1024 dimensions.

4.1.2 Gaussian Mixture Model

The s-th($1 \leq s \leq K$)center is used as initial means μs of the s- th Gaussian in GMM. Then the initial weights ws and co- variances σs are calculated from the means. Next, an EM algorithm is used to obtain maximum likelihood estimate of the three parameters, weights, means, and co-variances of all the Gaussian mixture distributions. A likelihood vector from all Gaussians is represented by Eq. (1).

$$Px = (ws\ ps(x|\mu s, \sigma s))|ks=1$$

$$=(w1\ p1(x|\mu 1, \sigma 1), w2\ p2(x|\mu 2, \sigma 2), \ldots, wK\ pK(x|\mu K, \sigma K)) \quad (1)$$

$$ps(x|\mu s, \sigma s) = 1/\sigma\sqrt{2\pi}\exp(-1/2(x - \mu/\sigma)^2)$$

Where x denotes a HOG-based feature vector at a key point, Px denotes the likelihood vector of feature vector x, and $ps(x|\mu s,\sigma s)$ denotes the probability value of x at the s-th Gaussian. For likelihood Vectors (Px,Py), where

$$Px = (ws\ ps(x \mid \mu s,\sigma s))|8s=1 \quad \text{and} \quad Py = (ws\ ps(y \mid \mu s,\sigma s))|8s=1$$

GMM- based feature representation by histogram of binary comparisons, as Eq. (2).

$$Fx, y = 8\Sigma^{s-1}\ [2^{s-1} * (P^s]$$
$$F = (Fx, y) \quad (2)$$

$$P(^s) = 1 \quad ; \text{if } ws\ ps\ (x \mid \mu s,\sigma s) \geq \text{if } ws\ ps\ (y \mid \mu s,\sigma s), \text{ and}$$
$$P(^s) = 0 \quad ; \text{if } ws\ ps\ (y \mid \mu s,\sigma s) > \text{if } ws\ ps\ (x \mid \mu s,\sigma s)$$

4.2 Character Stroke Configuration

In previously proposed method [7] stroke width consistency is used to detect scene text in complex background and achieve outstanding performance. Stroke is region bounded by two parallel boundary segments. Their orientation is regarded as stroke orientation and the distance between them is regarded as stroke width. The stroke configuration is estimated by synthesized characters generated from computer software. Character boundary and character skeleton are obtained by applying discrete contour evolution (DCE) [10] and skeleton pruning on the basis of DCE [11]. The accuracy of the skeleton position and stability of skeletons is guaranteed in this pruning method.DCE and skeleton pruning are invariant to deformation and scaling. Our estimate the stroke width and orientation on sample points of character boundary.

N points are sampled evenly from the polygon character boundary, with the polygon vertices reserved. In our experiment, we set N = 128. The number of points to be sampled on each side of the polygon boundary is proportional to its length. Secondly, stroke is contiguous part of an image that forms a band of a nearly constant width. We take b and its two neighboring sample points to fit a line when they are approximately collinear or else quadratic curve. Then the slope or tangent direction at b is used as stroke orientation. Characters are connected strokes with orientation. Thirdly, we calculate the skeleton-based stroke maps. At each boundary sample point, values of stroke width and orientation are compared with its neighboring points. Constituency of stroke width and orientation consistency 3 and π/8 respectively. Construct stroke section if sample points satisfy stroke related features. If not construct junction. These parameters are compatible with the synthesized character patches with size 128 × 128. While the other sample points, around

the intersections of neighboring strokes or the ends of strokes, compose junction sections of a character boundary.

4.2.1 Stroke Alignment Method

The basic structure of a character class can be described by the mean value of all stroke configurations from character samples of the class. Here, we estimate a mean value of stroke configuration so that it is able to handle various fonts, styles and sizes. Eq. (3) gives an objective function of stroke alignment.

$$E = \Sigma_i \, (D \, (\acute{S}, \, T_i \, (S_i \,)) + g(T_i \,)) \qquad (3)$$

$$D \, (Sm, \, Sn \,) = \sum \dot{p} \| Sm \, (p) - Sn \, (p) \|^2 \qquad (4)$$

D represents the distance between the stroke configurations of two character samples as Eq. (4), and $\acute{S}$ represents mean value of the stroke configuration s_i represents the transformation applied on the strokes of the i -th stroke configuration S_i.

5. RESULTS AND DISCUSSIONS

In what follows, we present a detailed evaluation of our method. We evaluate various components of the proposed approach to justify our choices. We compare our results with the best performing methods for the word recognition task.

Table -2: Accuracy rates of scene character recognition in ICDAR-2003 dataset compared with previously published results

ICDAR-2003 Dataset	AR
Ours	0.628
HOG+NN	0.515
SYNTH+FERNS	0.520
NATIVE+FERNS	0.640

Table -2: Accuracy Rates (AR) and False Positive Rates (FPR) of queried character classification in the three datasets

Dataset	AR	FPR
Char s 74K	0.726	0.078
Sign	0.868	0.0 75
ICDAR-2003	0.536	0.180

6. COMPARISON OF RESULTS

The experimental results in first Table show that our proposed descriptor outperforms the SYNTH+FERNS with AR 0.52 and comparable with NATIVE+FERNS having AR of 0.64.A character classifier is trained for each character class by using Chars74K samples, which is then evaluated over the three datasets to obtain the results. As shown in second table a character classifier is trained for each character class by using Chars74K samples, which is then evaluated over the three datasets to obtain the results.

7. CONCLUSION

Thus this paper achieves the objective of text extraction from image and displaying its information on android platform, w*ith the help of text extraction* algorithm.

It detects text regions from natural scene image/video, and recognizes text information from the detected text regions. Text understanding and text retrieval are respectively proposed to extract text information from surrounding environment. Character descriptor is effective to

extract representative and discriminative text features for both recognition schemes. To model text character structure for text retrieval scheme, we have designed a novel feature representation, stroke configuration map, based on boundary and skeleton. Quantitative experimental results demonstrate that proposed method of scene text recognition outperforms most existing methods.

REFERENCES:

1. Chucai Yi, "Scene Text Recognition in Mobile Applications by Character Descriptor and Structure Configuration,"IEEE Trans. Image Process., vol. 23, no. 7, July2014,pp.2972-2982**.**
2. D. L. Smith, J. Feild, and E. Learned-Miller, "Enforcing similarity constraints with integer programming for better scene text recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2011,pp.73–80.
3. C. Yi and Y. Tian, "Localizing text in scene images by boundary clustering, stroke segmentation, and string fragment classification," IEEE Trans. Image Process., vol. 21, NO. 9, PP. 4256–4268, SEP. 2012.
4. C. Yi and Y. Tian, "Text string detection from natural scenes by structure-based partition and grouping," IEEE Trans. Image Process., vol. 20, no. 9, pp. 2594–2605, Sep. 2011.
5. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2005, pp. 886–893.
6. T. de Campos, B. Babu, and M. Varma, "Character recognition in natural images," in Proc. VISAPP, 2009.
7. B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In CVPR, 2010.
8. M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. IJCV, 2010.
9. V. Kolmogorov. Convergenttree-reweighted message passing for energy minimization. PAMI, 2006.
10. C. Colombo, A. D. Bimbo, and P. Pala, Semantics in Visual Information Retrieval, IEEE Multimedia, 6 (3) (1999) 38-53.
11. T. Sato, T. Kanade, E. K. Hughes, and M. A. Smith, Video OCR for Digital News Archive, Proc. of IEEE Workshop on Content based Access of Image and Video Databases, 1998, pp. 52-60.
12. Atsuo Yoshitaka and Tadao Ichikawa, A Survey on Content-based Retrieval for Multimedia Databases, IEEE Transactions on Knowledge and Data Engineering, 11 (1) (1999) 81-93.
13. W. Qi, L. Gu, H. Jiang, X. Chen, and H. Zhang, Integrating Visual, Audio, and Text Analysis for News Video, Proc. of IEEE International Conference on Image Processing, 2000, pp. 10-13.
14. H. D.Wactlar, T. Kanade, M. A. Smith, and S. M. Stevens, Intelligent Access to Digital Video: The Informedia Project, IEEE Computer, 29 (5) (1996) 46-52.
15. A.Gupta, Y. Verma, and C.V. Jawahar. Choosing linguistics over vision to describe images. In AAAI, 2012.
16. D. Hoiem, A. Efros, and M. Hebert. Closing the loop on scene interpretation. In CVPR, 2008.
17. Anand Mishra Karteek Alahari C. V. Jawahar. Top-down and bottom-up cues for scene text recognition. pages 1–3.
18. A. Mishra, K. Alahari, and C. V. Jawahar, "Top-down and bottom-up cues for scene text recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2012, pp. 1063–6919.
19. S. Lu, L. Li, and C. L. Tan, "Document image retrieval through word shape coding," IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 11, pp. 1913– 1918, Nov. 2008