SCENE TEXT RECOGNITION IN MOBILE APPLICATIONS BY CHARACTER DESCRIPTOR AND STRUCTURE CONFIGURATION

Sathish Kumar Penchala[1], Pallavi S.Umap[2]

[1]Assistant Professor, Dept. of Computer Engineering, Dr. D.Y.Patil SOET., Lohegaon, Pune-47, Maharashtra, India
[2]ME 2nd year, Dept. of Computer Engineering, Dr.D.Y.Patil SOET., Lohegaon, Pune-47, Maharashtra India

Abstract

Camera-based scene images usually have complex background filled with non-text objects in multiple shapes and colors. The existing system is sensitive to font scale changes and background interference. The main focusof 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. Proposedsystem accurately extracts text from natural scene image in presence of background interference.The demo system gives us details of algorithm design and performance improvements of scene text extraction. It is ableto 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.*

--***--

1. INTRODUCTION

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.

First, 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]. Our algorithm design is compatible with the application of scene text extraction in smart mobile devices. An Android-based demo system is developed to show the effectiveness of our proposed method on scene text information extraction from nearby objects.

2. RELATED WORK

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. prior models of the appearance of each character, and prior models of the likelihood of each character string. These models can be learned using traditional vision techniques and statistical language modeling techniques [2]. 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

3.1Layout 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 most scene images, text strings are usually composed of characters with similar size and 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.

Table -1: Summary of the Previous Techniques

Title of Paper	Author/ Authors of paper	Previous Technique Used
Scene text recognition using similarity and a lexicon	J. J. Weinman, E. Learned-Miller	Combined Gabor-based appearance model.
Real-time scene text Localization And recognition	L. Neumann , J. Matas	Based on extremal region
Enforcing similarity Constraintswith integer programming	D. L. Smith, J. Feild, Learned-Miller	Based on SIFT
Document image retrieval ThroughWord shape coding	S. Lu, L. Li, C. L.Tan	Model the inner Character structure
character recognition in scene images with unsuper-vised feature learning	A. Coates et al.	extracted local features of character patches

4. STRUCTURE-BASED SCENE TEXT RECOGNITION

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.

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 1 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.

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

BOW model is applied to key points from all four feature detectors. This model is computationally efficient and resistant to intra-class variations. First, k-means clustering is performed on HOG features extracted from training patches. 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.

4.1.2 Gaussian Mixture Model

The *s*-th($1 \leq s \leq K$)center is used as initial means μ_s of the *s*-th Gaussian in GMM. Then the initial weights w_s and co-variances $\sigma_s$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).

$$\text{Px} = \sum_{s=1}^{k} w_s p_s(\text{x}|\mu_s, \sigma_s)\ (1)$$

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

Where *x* denotes a HOG-based feature vector at a key point, Pxdenotes the likelihood vector of feature vector *x*, and $p_s(x|\mu_s,\sigma_s)$ denotes the probability value of x at the s-th Gaussian. For likelihood Vectors (P_x,P_y), where

$$\text{Px} = \sum_{s=1}^{k} w_s p_s(x|\mu_s, \sigma_s) \quad \text{and} \quad \text{Py} = \sum_{s=1}^{k} w_s p_s(\text{y}|\mu_s, \sigma_s)$$

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

$$\text{Fx, y} = \sum_{s=1}^{k} 2^{s-1} \times\ (\text{Ps})\ (2)$$

$P^{(s)}$ = 1 ; if $w_s p_s(x|\mu_s,\sigma_s) \geq$ if $w_s p_s(y|\mu_s,\sigma_s)$, and
$P^{(s)}$ = 0 ; if $w_s p_s(y|\mu_s,\sigma_s) >$ if $w_s p_s(x|\mu_s,\sigma_s)$

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.

We 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 nearlyconstant width. We take b and its two neighboring sample points to fit a line when they are approximately collinear or else a quadratic curve. Then the slope or tangent direction at b is used as stroke orientation as shown in figure 2 below. 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.

$$\text{D(Am, An)} = \sum_p ||A_m(\text{P}) - A_n(\text{P})||^2, \text{E} = \sum_i (\text{D}(\bar{\text{A}}, T_i(A_\text{i})) + g(T_i)) \qquad (3)$$

D=distance b/w stroke configuration of samples.
A=mean value of stroke configurations
T_i = Transformations applied on strokes of i-th stroke configuration.
$g(T_i)$=Amplitude of transformation.

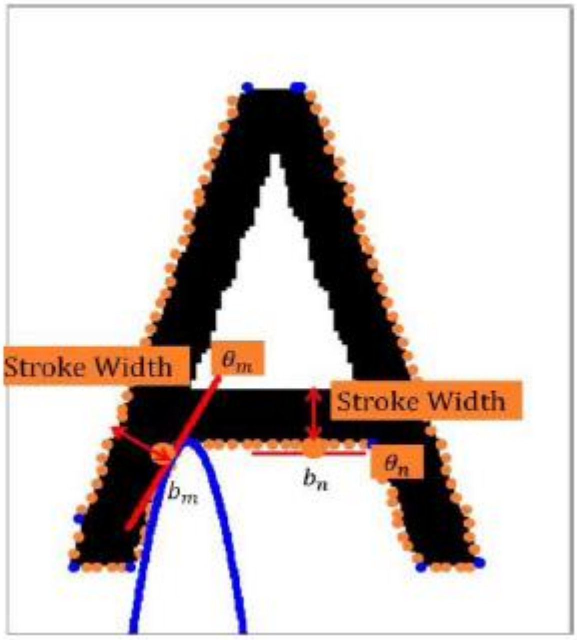

Fig -2: Stroke Orientations and stroke width denoted by red line and red double arrows resp.,$b_n$is approximately collinear ,while b_m fits a quadratic curve with neighboring point

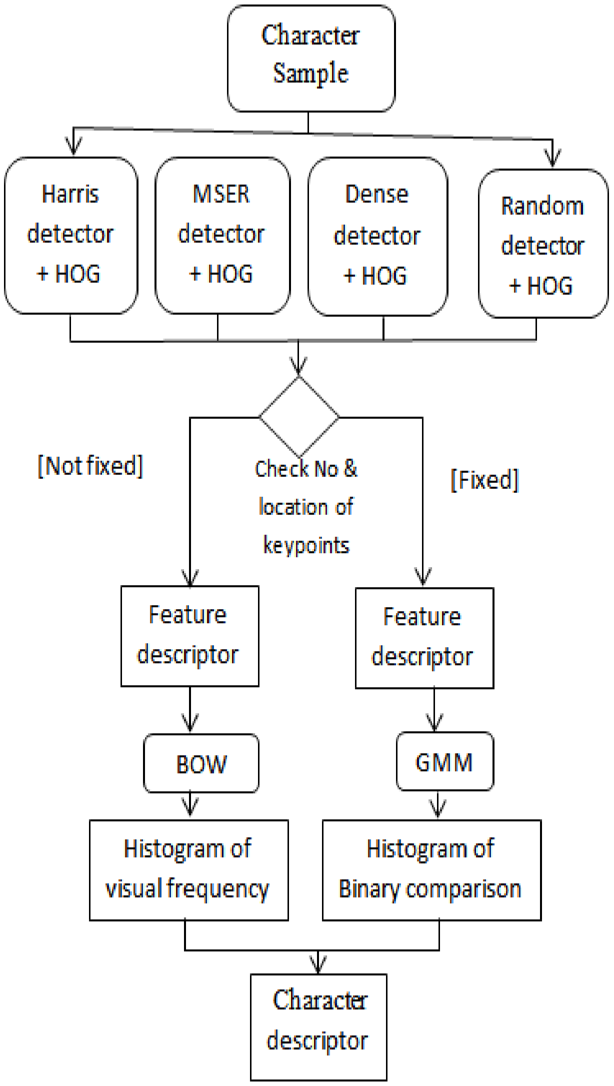

Fig -2: Flowchart of our proposed character descriptor

5. RESULTS AND DISCUSSIONS

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

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

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

Dataset	AR	FPR
Chars74K	0.726	0.078
Sign	0.868	0.075
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

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, July 2014, 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. and B. Triggs, “Histograms of oriented gradients for human detection,” in Proc. IEEEConf. Comput. Vis. Pattern Recognit., Jun. 2005, pp. 886–893.
[6]. P. Viola and M. J. Jones, “Robust real-time face detection,” Int. J.Comput. Vis., vol. 57, no. 2, pp. 137–154, 2004.
[7]. B. Epshtein, E. Ofek, and Y. Wexler, “Detecting text in natural scenes with stroke width transform,” in Proc. CVPR, Jun. 2010, pp. 2963–2970.
[8]. J. J. Weinman, E. Learned-Miller, and A. R. Hanson, “Scene text recognition using similarity and a lexicon with sparse belief propagation,”IEEETrans. Pattern Anal. Mach. Intell., vol. 31, no. 10, pp. 1733–1746, Oct. 2009.
[9]. J. Zhang and R. Kasturi, “Extraction of text objects in video documents: Recent progress,” in Proc. 8th IAPR Int. Workshop DAS, Sep. 2008, pp. 5–17.

[10]. L. J. Latecki and R. Lakamper, “Convexity rule for shape decomposition based on discrete contour evolution,” Comput.Vis. Image Understand., vol. 73, no. 3, pp. 441–454, 1999.

[11]. X. Bai, L. J. Latecki, and W.-Y. Liu, “Skeleton pruning by contourpartitioning with discrete curve evolution,” IEEE Trans. Pattern Anal.Mach. Intell., vol. 29, no. 3, pp. 449–462, Mar. 2007.

BIOGRAPHIES

Sathish Kumar Penchala, Assistant Professor in Computer Engineering at Dr. D.Y.Patil SOET, Lohegaon, Pune-47.

Pallavi **S. Umap,** is a Final year Post Graduate student,pursuing her degree Master of Engineering degree in ComputerNetworks at Dr. D.Y.Patil SOET., Lohegaon, Pune-47.