IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-3, Issue-4 E-ISSN: 2347-2693

Detection of Text Using Connected Component Clustering and Nontext Filtering

S. Elakkiya[1*] and T. Kavitha[2]

[1*,2]*Dept. of CSE, Periyar Maniammai University, Thanjavur*

www.ijcseonline.org

Received: April /02/2015 Revised: April/11/2015 Accepted: April/23/2015 Published: April/30/ 2015

***Abstract*—** Several methods have been developed for text detection and extraction to achieve accuracy for natural scene text and for multi-oriented text. However most of the methods use classifier to improve text detection accuracy. So this paper uses two machine learning classifiers one is to generate candidate region and the other filters nontext. Here connected components (CCs) in images are extracted by using the maximally stable extremal region algorithm. These extracted CCs are partitioned into clusters so that we can generate candidate regions. An AdaBoost classifier is trained to determine the adjacency relationship and cluster CCs by using their pair-wise relations. Since the scale, skew, and color of each candidate can be estimated from CCs, we can develop a text/nontext classifier for normalized images. This classifier will be based on multilayer perceptrons and we can control recall and precision rates with a single free parameter. Finally, the approach can be extended to exploit multichannel information and this method yields the state-of-the-art performance both in speed and accuracy.

***Keywords*—**Connected Component,Clustering, Extraction, Filtering.

I. INTRODUCTION

Human and computer interaction (HCI) attains wide attention due to the introduction of mobile devices equipped with high resolution digital cameras which are widely available in many activities. Among them, text detection and recognition in camera based images have been considered as very important problem. It is because the text data is easily recognized by machines and can be used in a variety of applications. In the same time due to complex background, and variations of font, size, colour and orientation, text detection in natural scene images has to be robustly detected before being recognized and retrieved. With the increasing popularity of mobile phones, text detection in natural scenes becomes a challenging task. This is because scene text images primarily suffer from photometric degradations and geometrical distortions so that many algorithms face the accuracy and/or speed (complexity) issues. To extract scene text information from camera-captured document images many algorithms and commercial optical character recognition (OCR) systems have been developed.

II. RELATED WORKS

Most text detection algorithms can be classified into two categories: texture-based and connected component (CC)-based method. Texture-based approaches view text as a special texture that is distinguishable from the document image background. In this features are extracted over a certain region and a classifier is used to identify the existence of text. Connected component based methods extract character candidates text from images by connected component analysis followed by grouping character candidates into text; additional operation is performed to remove false positives. Recently, Maximally Stable Extremal Regions (MSERs) based method is used, which can be categorized as connected component based method. Connected component analysis method is used to define the final binary images that mainly consist of text regions. After the CC extraction, CC-based approaches filter out non-text. Finally, CC-based approaches infer text blocks from the remaining CCs. This step is also known as text line formation, or text line grouping.

The other methods like region based method have focused on binary classification texts versus non text of a small image patch. Mainly they focused on the problems to determine whether a given patch is a text region or not, many experimental results shows that it is efficient therefore its performance worse compared with CC-bases approaches.CC-based method extract and normalize text regions by processing only CC-level information and they focused on problems like to extract text-like CCs, filter out non text CCs and to infer text blocks from CCs.

Some of the other works are

2.1 Robust text detection in natural images with edge-enhanced maximally stable extremal regions

In this paper a novel CC-based text detection algorithm is used which employs Maximally Stable Extremal Regions (MSER) as our basic letter candidates. Despite the properties, MSER has been reported to be sensitive to image blur and it also allow to detect small

letters in images of limited resolution, the properties of canny edges and MSER are combined in edge-enhanced MSER. Further to generate the stroke width transform image of these regions using the distance transform to efficiently obtain more reliable results. The geometric as well as stroke width information are then applied to perform filtering and pairing of CCs.

The main advantage of using this MSER-based methods over traditional connected component based method is able to detect most characters even when the image is in low quality (low resolution, strong noises, low contrast, etc.

2.2 Detecting Text in Natural Scenes with Stroke Width Transform

In this paper a novel image operator tries to find the value of stroke width for each image pixel to demonstrate its use on the task of text detection in natural images. Then the suggested operator is either local or data dependent, which makes it fast and robust enough to eliminate the need for multi-scale computation or scanning windows and an extensive testing shows that it outperforms the other algorithms. The algorithm simplicity allows detecting texts in many fonts and languages.

2.3 A classification architecture based on connected components for text detection in unconstrained environments

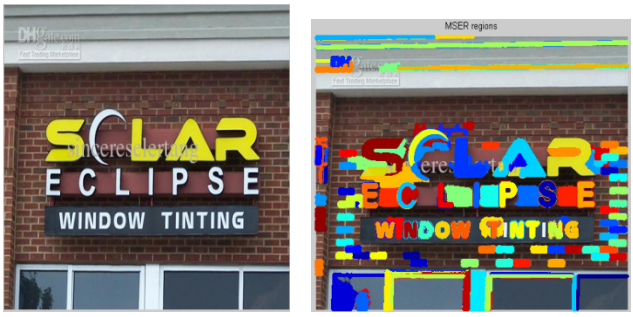
In this the method is based on segmenting the image in Connected Components (CC) and classifying them in text and not-text with classification architecture. The classification architecture is based on Regularized Least Squares (RLS).In the classification phase they train a classifier on a training set of positive (text connected components) and negative (non text) examples described according to the described CCfeatures. Since the CC-features are heterogeneous the main issue is how to combine the features needs to be addressed. An approach is to build a global feature vector after all features are normalized to a common range of values. The major disadvantage with this approach is computational which deals with an object detection problem aims minimize the number of evaluations for each analyzed image region.

III. PROPOSED SYSTEM

We proposed a system which overcomes some of the problems mentioned in the existing system. It is based on machine learning perspective. It consists of three parts which is illustrated in Fig.1 and they are listed as candidate generation, candidate normalization, and non-text filtering. The detailed information of various components of our proposed system is explained below.

IV. CANDIDATE GENERATION

In the process of candidate generation, the extraction of Connected Components (CCs) is done followed by the partitioning of the extracted CCs into clusters which is based on an adjacency relation classifier. So in our CC extraction method three steps we used three following steps (i) to build training samples, (ii) to train the classifier, and (iii) to use that classifier in our CC clustering method. Thus at the end of this process each connected components are filled with different colors as in fig 2.



(a) **(b)**

Figure 2: (a) Input Image (b) Connected Component after MSER extraction

a) Cc Extraction:

In CC extraction MSER algorithm is used which extract number of co-variant regions from an image. This algorithm can be used as a process to find local binarization results that are stable over a range of thresholds, and allows us to find most of the text component. The MSER algorithm yields CCs that are either darker or brighter than their surroundings. Many CCs are overlapping due to the properties of stable regions. If the sequence of threshold images lt with frame t corresponding to threshold t, we obtain a black image, then white spots corresponding to local intensity minima will appear then grow larger. Then the white spots will eventually merge, until the whole image is white and the set of all connected components in the sequence is equal to set of all extremal regions.

b) Building Training Sets:

Training sets are built based on pair wise relations between CCs. The following are the cases for a CC pair.

1) $Ci \epsilon T, Cj \epsilon T, Ci \sim Cj$
2) $Ci \epsilon T, Cj \epsilon T, Ci \sim Cj, t(Ci) = t(Cj)$
3) $Ci \epsilon T, Cj \epsilon T, Ci \sim Cj, t(Cj) \neq t(Cj)$

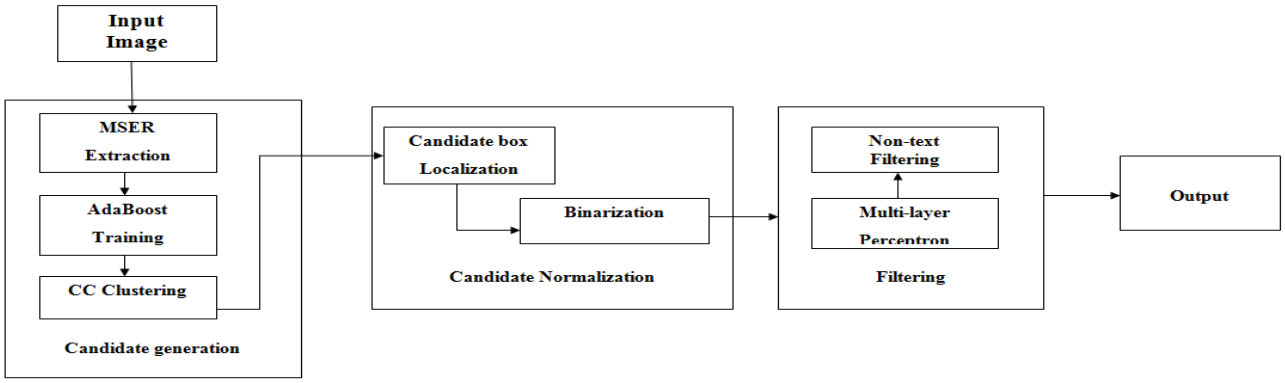


Figure 1: Architecture of Proposed System

4) Ci ϵ T, Cjϵ N
5) Ci ϵ N, Cjϵ N.

Here Ci and Cj represent the connected components, T and N represents the text and nontext sets. t(Ci) and t(Cj) represents the text line for that particular connected component. Ci ~ Cj represents Ci is adjacent to Cj.

Based on the above observations, we build training sets. Corresponding to the case (1) and corresponding to the case (3) or (4) a positive set and a negative set are built respectively by gathering the samples. Samples corresponding to other cases are discarded.

c) AdaBoost Learning

Using the samples we train an AdaBoost classifier which gives us information whether (Ci, Cj) is adjacent or not. It is an algorithm for constructing a strong classifier as linear combination. Strong classifier is built by the combinations of many weak classifiers.

Mainly AdaBoost starts with a uniform distribution of "weights" over training examples and then obtain a weak classifier for n iterations from the weak learning algorithm. The weight of the training examples that were misclassified is increased. The process is repeated till the end of the iteration. In the end, a linear combination of the weak classifiers obtained at all iterations to obtain a strong classifier.

d) CC Clustering:

The AdaBoost algorithm yields a function and that function is used in clustering decisions. With that we can find all adjacent pairs by evaluating all possible pairs. Based on these adjacency relations, set of connected components is partitioned in to a set of clusters. After clustering, we have discarded clusters having only one CC.

V. CANDIDATE NORMALIZATION

After cc clustering we have a set of clusters. In this we process normalization of corresponding regions performed for reliable text/non-text classification.

a) Geometric Normalization:

First we localize the corresponding region. Then by normalization process we can geometrically transform the image into a standard form and shape. After we obtain the corresponding regions, we approximate the shape of the region with the text boxes in the form of parallelograms. Here the left and right sides of the parallelogram are made parallel to the y-axis. Then, by applying an affine mapping we perform geometric normalization that transforms the corresponding region to a rectangle. During the transformation, we use a constant target height (48 pixels in experiments) and preserve the aspect ratio of the box.

b) Binarization of the images:

Now we perform the binarization of all the corresponding regions .There is only two possible values for each pixel in a binary image. Typically the two colors used for a binary image are black and white. Binarization can be done based on MSER, but might not be efficient because the result of MSER miss some character components or yield noisy regions due to the blur. Then we have to store the point information of all CCs for the MSER-based binarization. Therefore binarization is done separately by estimating text and background colors.

VI. NONTEXT FILTERING

To get final results a text/nontext classifier that rejects nontext blocks among normalized images is developed. The main challenge of the approach is the variable aspect ratio. There is one possible approach to solve this problem is to split the normalized images into patches covering one of the letters and develop a character/non-character classifier. However, character segmentation is not an easy problem. So we split a normalized block into overlapping squares and develop a classifier that assigns a textness value to each square block. Finally, the decision results for all square blocks are integrated so that the original block is classified.

a) Feature Extraction from a Square Block

Our feature vector is based on mesh and gradient features. We divide each square into 4 horizontal and four vertical blocks which extract features. For horizontal block we consider

1) The number of white pixels,
2) The number of vertical white-black transitions,
3) The number of vertical black-white transitions

As features, and features for a vertical block is similarly defined.

b) Multilayer Perceptron

For the training, we need normalized images. For this we apply our algorithm presented in the previous sections i.e., candidate generation and normalization algorithms to the training images. Then, we manually classified them into text and nontext. Here we discarded some images that show poor binarization results, after that text block images and nontext block images are collected. However, we found that more negative samples are needed for the reliable rejection of nontext components and collected more negative samples by applying the same procedure to images that do not contain any text.

These text/nontext images are divided into squares and we have trained a multi-layer Perceptron for the classification of square patches. Later we use one hidden layer consisting of 20 nodes and set the output value to +1 for text samples and 0 otherwise. To help the learning, input features are normalized.

(a) (b)

Figure 3: (a) Detected text highlighted with Rectangle (b) Extracted Text

VII. PERFORMANCE EVALUATION

We evaluate the performance of our technique based on its precision and recall rates. Precision rate takes into consideration of the false positives, which are the non-text regions in the image and have been detected by the algorithm as text regions. Recall rate takes into consideration of the false negatives, that is the text words in the image, and which is not been detected by the algorithm. So, precision and recall rates are useful as measures to determine the accuracy of each algorithm in locating correct text regions and eliminating non-text regions.

Our method based on machine learning classifiers will efficiently locate the text regions. Their accuracy can be measured based on their precision rate and evaluation rate.

VIII. CONCLUSION

In this paper, the algorithm used for scene text detection based on machine learning methodology where the system will be tested and trained based on data sets.

Two machine learning classifiers used, in that one helps in the generation of candidate word regions and the other filters out nontext ones efficiently. Then its efficiency is calculated based on the precision rate and recall rate.

REFERENCES

[1] K. Jung, "Text information extraction in images and video A survey," Pattern Recognit., vol. 37, no. 5, pp. 977–997, May 2004.

[2] S. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, "ICDAR 2003 robust reading competitions," in *Proc. Int. Conf. Document Anal. Recognit,* 2003, pp. 682 687.

[3] S. Lucas, "Icdar 2005 text locating competition results," in *Proc. Int. Conf. Document Anal. Recognit.* 2005, pp. 80–84.

[4] Shahab, F. Shafait, and A. Dengel, "ICDAR 2011 robust reading competition challenge 2: Reading text in scene images," in *Proc. Int. Conf. Document Anal. Recognit,* 2011, pp. 1491–1496.

[5] Hyung Il Koo and Duck Hoon Kim, "Scene Text Detection via Connected Component Clustering and Nontext Filtering," IEEE Trans. Image Process., vol. 22, no. 6, pp.2296-2305, June. 2011.

[6] X. Chen and A. Yuille, "Detecting and reading text in natural scenes," in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2004, pp. 366–373.

[7] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2963–2970.

[8] H. Chen, S. Tsai, G. Schroth, D. Chen, R. Grzeszczuk, and B. Girod, "Robust text detection in natural images

with edge-enhanced maximally stable extremal regions," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 2609–2612.

[9] X. Chen and A. Yuille, "A time-efficient cascade for real-time object detection: With applications for the visually impaired," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Workshops, Jun. 2005, pp. 1–8.

[10] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *Ann. Stat.*, vol. 28, no. 2, pp. 337–407, 1998.