www.ijiarec.com

ISSN:2348-2079

Volume-5 Issue-1

International Journal of Intellectual Advancements and Research in Engineering Computations

Detection and recognition of the text in image using cloud computing clustering

1. S. Lalitha[1], 2. Dr. N. Shanthi [2], 3. R. Umamaheswari [3],

1 & 3 Research Scholars, Computer Science and Engineering, Gnanamani College of Technology.

2. Professor & Dean / CSE, Nandha Engineering College, Erode

Mailid: 1.shemaait@gmail.com , 3. umait1978@gmail.com

Abstract— In the new generation the artificial intelligence is going to control everything in the world. In the existing system, the character detection in the image is done by connected component (CC) based approach.CC based approach is computational complexity, based on the pixel difference between the text and background of the image text is detected. Text images usually suffer from photometric degradation as well as geometric distortions. The proposed system will extract and recognize the characters too. This process is going to do with the help of training the fonts of alphabets and numbers which are stored in database. Gabour filter is used for the reduction of noise. Here the number plates of the vehicles are logged with time and date can be used in the toll gate. For the process of performance evaluation the similarity factor is going to done and 95% accuracy is expected in the detection of the numbers. From the detected text it can converted to audio for visually impairing people. And it is useful in the toll gates for automatic number entry. The project can be simulated by using MATLAB software.

Index Terms— Adaboost algorithm, Connected Component (CC), Gabour filter, Maximally Stable Extremal Region (MSER), Multilayer Perceptron.

I. INTRODUCTION

Text detection and recognition in camera captured images have been considered as very important problems in computer vision community. It is because text information is easily recognized by machines and can be used in a variety of applications. Some examples are aids for visually impaired people, translators for tourists, information retrieval systems in indoor and outdoor environments, and automatic robot navigation. Although there exist a lot of research activities in this field, scene text detection is still remained as a challenging problem. This is because scene text images usually suffer from photometric degradations as well as geometrical distortions so that many algorithms faced the accuracy and/or speed (complexity) issues.

II. CC METHOD

After the CC extraction, CC-based approaches filter out non text CCs. In the end, features such as “aspect ratio,” “the number of holes in a CC,” and “the variance of the stroke width within each CC” were employed in. In conditional random fields (CRFs) were adopted in order to consider binary (relational) features as well as unary features. In a neural network was used to filter out non text components. Among a number of CC extraction methods, we have adopted the MSER algorithm because it shows good performance with a small computation cost. This algorithm can be considered as a process to find local binarization results that are stable over a range of thresholds, and this property allows us to find most of the text components. The MSER algorithm yields CCs that are either darker or brighter than their surroundings.

The main aim of CC grouping is to group adjacent characters detected in the previous steps into separated meaningful words and further reject false positives. Figure 5.3 shows the local properties between two connected components. Based on the observation that characters in the same word usually share some similar properties, such as intensity, size, stroke width etc., this valuable information can be utilized in CC grouping.

A. Geometric Normalization

Given $w_i \in W$, first localize its corresponding region. Even though text boxes can experience perspective distortions, approximating the shape of text boxes with parallelograms whose left and right sides are parallel to y-axis. This approximation alleviates difficulties in estimating text boxes having a high degree of freedom (DOF): we only have to find a skew and four boundary supporting points. To estimate the skew of a given word candidate w_k , we build two sets:

$T_k = \{t(c_i) | c_i \in w_i\}$

$B_k = \{b(c_i) | c_i \in w_i\}$ (1)

Where $t(c_i)$ and $b(c_i)$ are the top-center point and the bottom center point of a bounding box of c_i , respectively. For every pair in B_k and T_k , the slope of a line connecting the pair is discretized into one of 32 levels in $[-\pi/8, \pi 8]$, and each pair votes for the skew angle. After voting, the most common angle is considered as a skew. Then, we perform geometric normalization by applying an affine mapping that transforms the corresponding region to a rectangle.

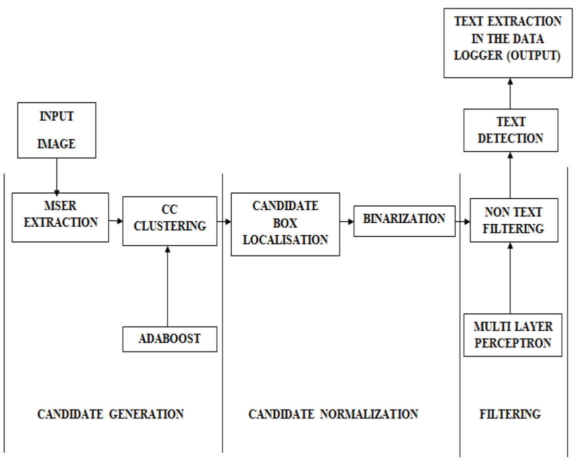

Fig.1.Block diagram of text extraction

III. TEXT/NON TEXT CLASSIFICATION

Developing a text/non text classifier that reject non text blocks among normalized images. In this classification, do not adopt sophisticated techniques such as cascade structures, since the number of samples to be classified is usually small. One possible approach to 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. Rather, split a normalized block into overlapping squares and develop a classifier that assigns a textness value to each square block. Finally, decision results for all square blocks are integrated so that the original block is classified.

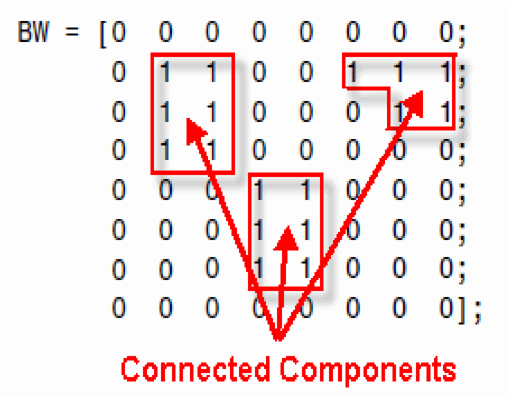

Fig.2 .showing connected component

Our CC extraction algorithm is the maximally stable extremal region (MSER) algorithm that is invariant to scales and affine intensity changes, and other blocks in our method are also designed to be invariant to these changes. These invariance allows us to exploit multi-channel information: we can apply our method to multiple channels at the same time and treat their outputs as if they are from a single source. In this way, we are able to detect text that are not salient in luminance channel images. Finally, CC-based approaches infer text blocks from the remaining CCs. This step is also known as text line aggregation, text line formation, or text line grouping. Interestingly, many methods were based on similar rules. For example, the height ratios between two letters and color difference have been used in a number of methods. Although CC-based approaches have shown better performance than region-based ones, they usually suffer from the computational complexity. It is because their performances depend on the quality of CCs and they adopted sophisticated CC extraction and filtering methods.

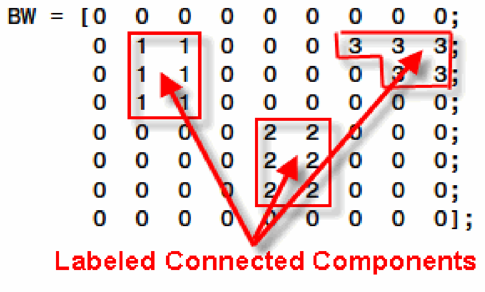

Fig.3.Connected Component of different pixel intensity

B. *Multilayer Perceptron Learning*

some images showing poor binarization results, and collected 676 text block images and 863 nontext block images. Finally, we have 3,568 non text images.

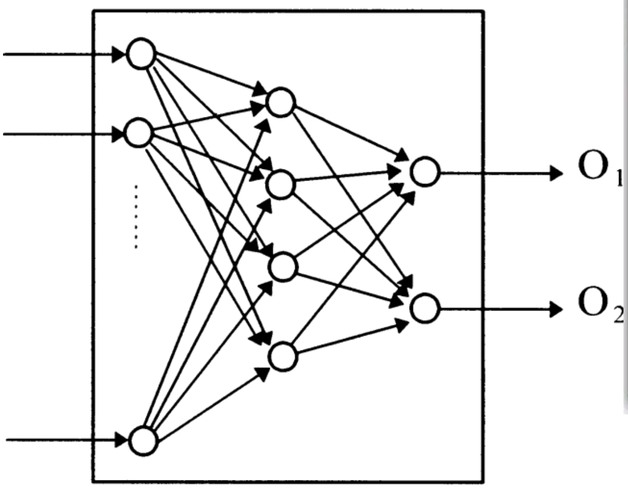

Fig.4.A sub network for the modular neural network

Multi-layer perceptron is trained for the classification of square patches, 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. Multilayer perceptron is a neural network used to compare the input image with the already trained texts then if the texts are matched then the given input text is detected as the output. Through the multilayer perceptron we can select as required input text for the comparison. Already trained some font styles are present in multilayer perceptron. The performance measure used for text detection, which is easier to define than for localization and extraction, is the detection rate, defined as the ratio between the number of detected text frames and all the given frames containing text. Measuring the performance of text extraction is extremely difficult and until now there has been no comparison of the different extraction methods. Instead, the performance is merely inferred from the OCR results, as the text extraction performance is closely related to the OCR output.

IV. RESULTS

The input is given as the color image here the number plate of the vechicle is shown.

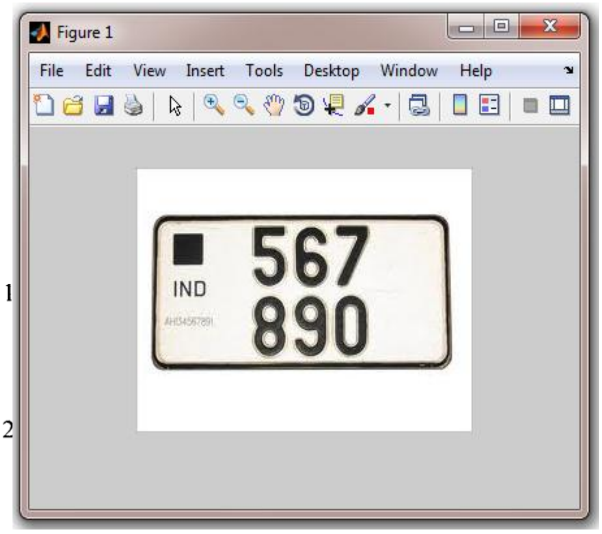

Fig.5.input image

Given geometrically normalized images, we build binary images. In many cases, MSER results can be considered as binarization results. Perform the binarization separately by estimating text and background colors. It is because (i) the MSER results may miss some character components and/or yield noisy regions (mainly due to the blur) and (ii) we have to store the point information of all CCs for the MSER-based binarization. Consider the average color of CCs as the text color. For the generation of candidates, we extract CCs in images and partition the extracted CCs into clusters, where our clustering algorithm is based on an adjacency relation classifier. Among a number of CC extraction methods, we have adopted the MSER algorithm because it shows good performance with a small computation cost. This algorithm can be considered as a process to find local binarization results that are stable over a range of thresholds, and this property allows us to find most of the text components. The MSER algorithm yields CCs that are either darker or brighter than their surroundings. The characters in the caption text appear in clusters and usually lie horizontally, although sometimes they can appear as non-planar texts as a result of special effects. This does not apply to scene text, which can have various perspective distortions.

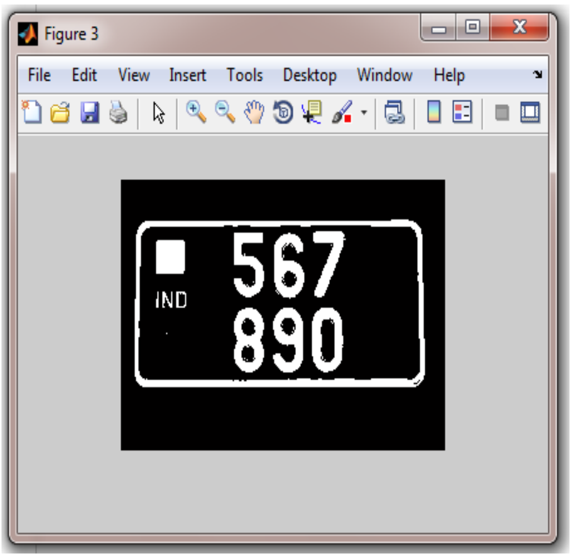

Fig.6.binarized image

In artificial intelligence text extraction most important thing for automation process. Connected Component is easy method to extract the image useful at the toll gate.

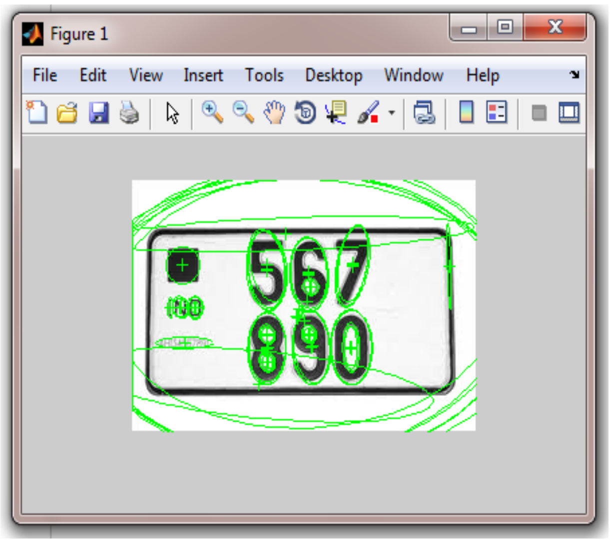

Fig.7.randomly localized image

The text/character like areas are detected as random boundary lines. The whole image is considered randomly form that the text like areas are detected. The image shown above is the input image with the text (figure-5).Image 6, shows binarized image. The randomly localized texts are localized which is shown in figure-7.the detection of the separate text is shown in the image is shown in the figure-8. Developing a text/non text classifier that reject non text blocks among normalized images. In this classification, do not adopt sophisticated techniques such as cascade structures, since the number of samples to be classified is usually small.

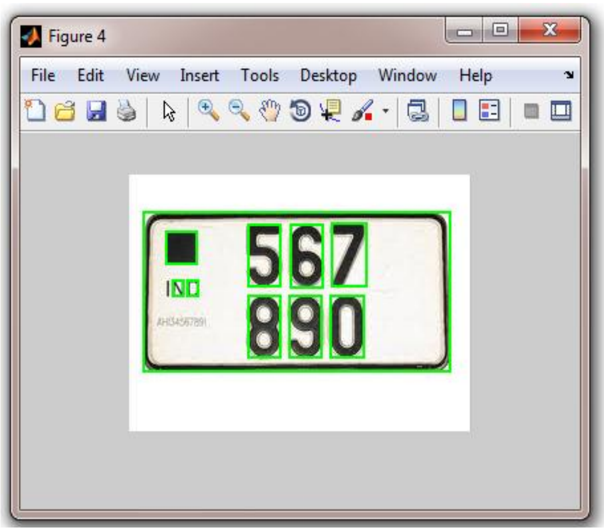

Fig.8.detected text image

The above image shows the text alone detected areas by boundary lines shown in green lines.

V. Proposed Method

The different steps of our approach are as follows. Step 1: Image Preprocessing-If the image data is not represented in YUV color space, it is converted to this color space by means of an appropriate transformation. After that, luminance value thresholding is applied to spread luminance values throughout the image and increase the contrast between the possibly interesting regions and the rest of the image. Step 2: Edge Detection-This step focuses the attention to areas where text may occur. As a result, all character pixels as well as some non-character pixels which also show high local color contrast are registered in the edge image. In this image, the value of each pixel of the original image is replaced by the largest difference between itself and its neighbors (in horizontal, vertical and diagonal direction). Despite its simplicity, this procedure is highly effective. Finally, the contrast between edges will be increased by means of a convolution with an appropriate mask. Step 3: Detection of Text Regions- The horizontal projection profile of the edge image is analyzed in order to locate potential text areas. Since text regions show high contrast values, it is expected that they produce high peaks in horizontal projection. Step 4: Enhancement and Segmentation of Text Regions- First, geometric properties of the text characters like the possible height, width, and width to height ratio are used to discard those regions whose geometric features do not fall into the predefined ranges of values.

All remaining text candidates undergo another treatment in order to generate the so-called text image where detected text appears on a simplified background. The binary edge image is generated from the edge image, erasing all pixels outside the predefined text boxes and then binarizing it. This is followed by the process of gap filling.

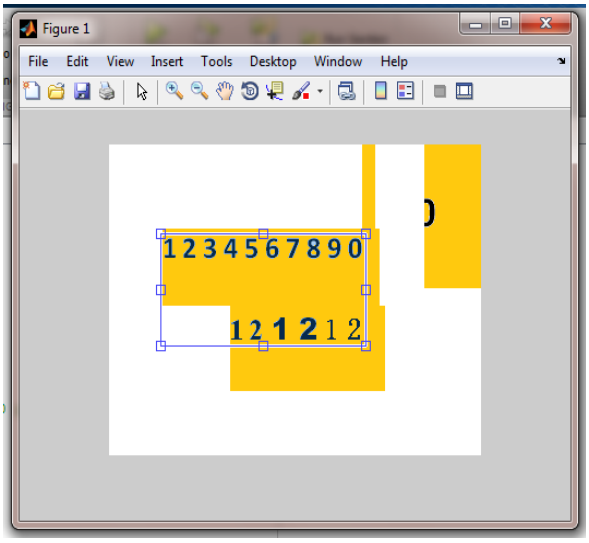

Fig.10.cropped input image(for numericals)

If one white pixel on the binary edge image is surrounded by two black pixels in horizontal, vertical or diagonal direction, then it is also filled with black. The gap image is used as a reference image to refine the localization of the detected text candidates. Text segmentation is the next step to take place. Then, the segmentation process concludes with a procedure which enhances text to background contrast on the text image.

The number plate of the vehicle contains both numbers and alphabets o we have train the classifier for numerical character and alphabets. It is tedious to train all type of font styles. So here training the classifier for only particular font styles. If the input contains another font styles it will not extract the same text in the image at the output.

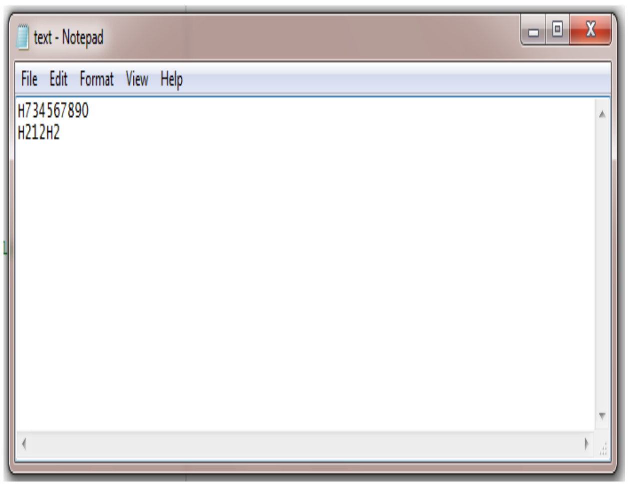

Fig.11.extracted text from the input image (numericals**)**

The above diagram shows the extraction of numerical numbers from the input image. Numerical characters are trained in the Calibri and Arial black font style. Alphabets are trained in times new roman

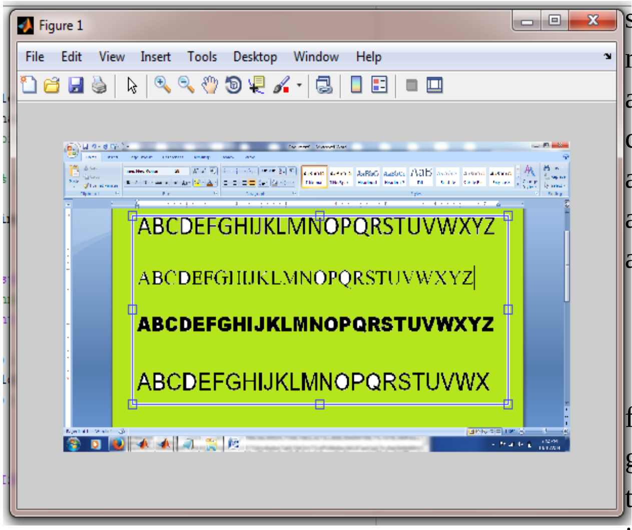

Fig.12.cropped input image (for alphabets)

The above diagram shows the input image for alphabets which is used for training of classifiers. Alphabets are trained as A to Z in Arial Black (bold), Times New Roman, and Calibri. If other than this format is given as input text extraction is not in accurate. Input image is given with the cropping option to select only the particular area. Output is produced only if trained font style is given as the input.

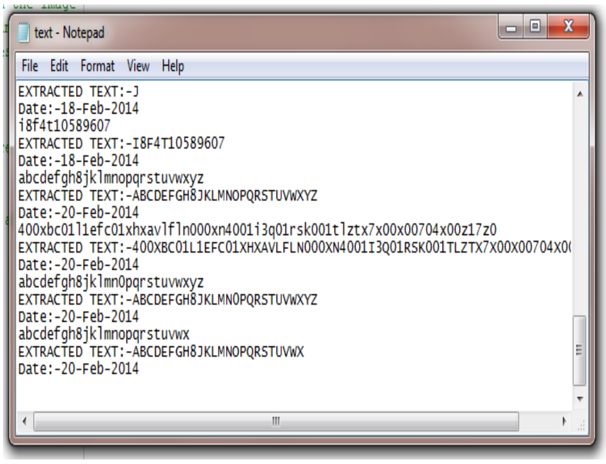

Fig.13. Extracted text for the above input image with date

VI. APPLICATIONS

The text information extraction system is used in many applications like including document analysis, vehicle license plate extraction in the toll gate. There has already been a lot of work done on vehicle license plate and container plate recognition. Although container and vehicle license plates share many characteristics with scene text, many assumptions have been made regarding the image acquisition process (camera and vehicle position and direction, illumination, character types, and color) and geometric attributes of the text, Robotic movement, automatic car driving, technical paper analysis and object-oriented data compression.

VII. FUTURE WORK

The extracted text from the image can be used for toll gate automatic exit of the vehicle if they got ticket for both entry and exit. By developing this project into real time application as hardware it can be useful in the toll gate which reduces the traffic and also saves time.

VIII. CONCLUSION

The proposed method has presented a novel scene text detection algorithm based on machine learning techniques. To be precise, we developed two classifiers: one classifier was designed to generate candidates by connected component method and the other classifier was for the filtering of non text candidates which is easy to detect the text from the images. Even though a large number of algorithms have been proposed for text extraction, no single method can provide satisfactory performance in all the applications.

REFERENCES

[1] Hyung Il Koo, *Member, IEEE, and Duck Hoon Kim, Member, IEEE* Scene Text "*Detection via Connected Component Clustering and Nontext Filtering*", IEEE transactions on image processing, *vol. 22, no. 6, june 2013*

[2] K. Jung, "*Text information extraction in images and video: A survey,"*Pattern

Recognit., *vol. 37, no. 5, pp. 977–997, May 2004.*

[3] J. Liang, D. Doermann, and H. Li,*"Camera-based analysis of text and documents: A survey,"* Int. J. Document Anal. Recognit.*, vol. 7, nos. 2–3,pp. 84–104, 2005.*

[4] X. R. Chen and A. L. Yuille, *"Detecting andreading text in natural scenes,*" in *Proc. IEEE Conf. Computer Vision and Pattern Recognition(CVPR'04)*, Washington, DC, *2004, pp. 366–373.*

[5] R. Casey and E. Lecolinet, *"A survey of methods and strategies in character segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 18, no. 7, pp. 690–706, Jul. 1996.*

[6] I.-S. Oh and C. Y. Suen, *"Distance features for neural network-based recognition of handwritten characters,*" *Int. J. Document Anal. Recognit.,vol. 1, no. 2, pp. 73–88, 1998.*

[7] P. Viola and M. Jones, "*Rapid object detection using a boosted cascade of simple features,*" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2001, pp. 511–518*.