Extraction Characters from Scene Image based on Shape Properties and Geometric Features

Abdel-Rahiem A. Hashem
Mathematics Department
Faculty of science
Assiut University, Egypt
University of Malaya, Malaysia

Mohd. Yamani Idna Idris
Faculty of Computer Science
and Information Technology
University of Malaya, Malaysia

Moumen T. El-Melegy
Electrical Engineering
Department
Assiut University, Egypt

ABSTRACT

Text extraction from scene images is a defy subject in light of low resolution, complex background and textual style/text size varieties. In this paper, we design a scheme to detect text based on shape features like Euler Number, a number of pixels for each region which candidate to be a character and vertical distances as a geometric feature between these regions. We divide these features into base features to collect the text regions, and the other features as a filter to discard the non-text regions. We use some threshold with the features either to extract text regions or to discard non-text regions. The proposed method outperforms some existed method through the basis metric.

Keywords

Scene text, Shape properties, Connected-components analysis.

1. INTRODUCTION

One of the most important objectives in computer vision is understanding images. Faces, people, animals, and text are commonly elements found in image but the text is the most useful thing among the content of the images so the major objective is to decide if there is text in a given image, what's more, if there is, to identify, confine, and retrieve it [1]. In by and large the primary two stages are detection and recognition. The sub-ventures in detection are localization and verification. The sub-ventures in recognition are segmentation and recognition as in Fig 1.

The sub-step verification can be incorporated with the sub-step localization and the sub-step segmentation can be coordinated with the progression recognition [1]. Most researchers arrange techniques used to extract texts in images into three classes; connected component-based techniques, texture-based techniques and edges based techniques [2]. Every classification of these strategies has a few burdens; connected component-based techniques are not robust in light of the fact that they be accepted when text pixels belong to the same region having the same features like color and intensity. Texture-based techniques might be inadmissible for small textual fonts and poor contrast text. Edge-based techniques give a false alerts more than different techniques and are not powerful for complex background images [3].

There are two fundamental classes of text; graphical text and scene text. Graphical text is text added to the image or video after they are caught, for example, inscriptions and marks. Scene text exists as an original content of images when it is specifically caught by a camera, for example, road names and movement signs.

The greater key of many techniques is binarization step, the binarization process is vital process among all processes which comprise the strategy, this is because of as binarization is strong and obvious, and the results of text detection also were strong and obvious.

In this paper, the proposed technique find horizontal and skewed text in light of the fact that a considerable lot of existed strategies did not make progress in finding the skewed text from scene images.

Whatever is left of this paper arranges as follow; section 2 we give the general brief survey of past work. Section 3 gives the past business related to the proposed technique. In section 4 we clarify the proposed system. The test and Experimental Results in section 5. We give the discussion and conclusion in section 6 with some future work.

2. LITERATURE REVIEW

Albeit connected component analysis is utilized to define the regions, Jie Yuan and et al, [4] utilize MSER detector for the same purpose, their proposed method made out of three stages. The initial step basically utilizes the MSER identifier to identify all candidate text regions. Despite the fact that the distinguished MSERs have consistent intensity in themselves, they are separated from each other. In the second step, they design a distance metric to measure the similarity between regions. Hopeful these regions merged to form text lines. At last step, a filter is utilized to dispose of non-content lines.

Kita and Toru [5] present a new approach which contains 3 stages; the initial step, generation tentatively binarized images via K-means clustering algorithm. The second step, applying

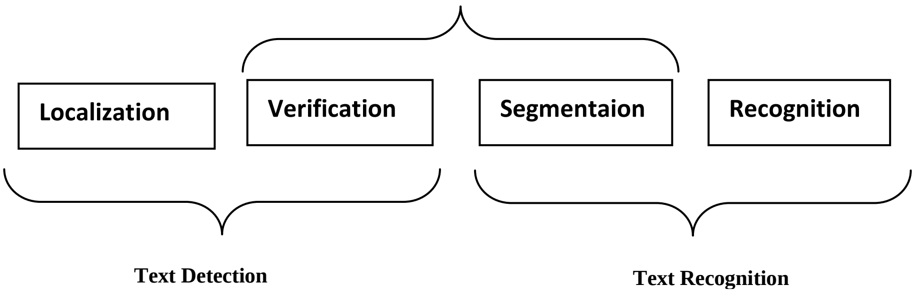

Fig. 1: Main steps for recognition text from scene images.

support vector machine to check if each binarized image represents a character region or no using a measure called “character-likeness”. The third step, choice single binarized image with the greatest level of "character- likeness” as an ideal binarization result.

Although edge-based methods are more efficient but these methods face a problem when the edges are not strong under influence of shadow or dim edge [6]. Shiva and et al, [3], [7] attempt to decrease false alarms by utilizing edge elimination rule depending on straightness properties of boundaries. What's more of utilizing some heuristic rules to discover applicant text pieces and after that they utilized similar rules for segmenting complete text parts in the image. These rules depending on the arithmetic mean filter and median filter. Shiva and et al. [8], proposed a system to identify and recognition bib number/text in Marathon natural images. The proposed strategy investigates histogram of gradients features alongside the support vector machine classifier for detect upper body in images, then Grab Cut technique is utilized for extract this text from this part of Image.

As we mentioned that binarization step is a center stride in many strategies handle text extraction, Sue Wu and Adnan Amin [9] present their approach contingent upon thresholding to change image into a binary image. The proposed strategy depends on two phases. On the main stage, global thresholding is utilized to recognize the locale/joint regions of the image (to run connected component analysis over the image). The second stage is to perform thresholding on areas to separate parts of characters.

3. RELATED WORK

In spite of the fact that there are research depend just on one of the three classes we specified above, however, current research tends to consolidate these classifications together, this is a result of retaining the benefits of these strategies.

In our previous work, we use statistical classifier named Naïve Bayes to classify image pixels into text pixels and non-text pixels, then we use connected-component analysis to compute area of all regions, then based on some heuristic rules on the number of pixels we retain text-region and discard non-text regions, one of the most disadvantages in our previous work is the depending on a single feature which was the number of pixels of each region detected by connected components [10]. Matias Valdenegro and et al [11], introduce their technique combining Maximum stable Extremal Regions (MSER) with a histogram of stroke width (HSW) feature.

After they use MSER to extract regions, they use HSW to classify these regions into text regions and non-text regions. They use two of shape features to normalize the stroke width values. This method does not adopt a single language to detect, but it is prepared to discover text from different scripts, this is a drawback because of the characteristics of the letters differ from one language to another, Also, its recall metric was reduced because of using the separated verification module to decrease the false positives. Liu C and et al, [2] Incorporate edge-based and texture-based strategy and proposed their approach which comprises of three stages; the first is getting four edge maps utilizing Sobel edge in four unique directions, then in the second step they compute the texture feature at every pixel from the edge maps utilizing the six elements, mean, standard deviation, energy, entropy, Interia, local homogeneity and correlation of edge maps. The k-means algorithm is applied to detect the initial text candidates. In the last stage, they utilize some of experimental rules to refine the candidates from previous stage. This method is not effective in the strong illuminations varieties and text distortion.

Albeit numerous OCR frameworks work admirably on reported images under controlled environment, they didn't give great outcomes in scene text images [12]; this is a direct result of unacceptable binarization results of scene images. In their work [12], Shi C and at al plan framework to localize and recognize the characters, in localization stage, they utilize MSER to recognize candidate text areas, then apply a tree-organized character models on these locales to eliminate false positives and find missing characters. Their method may fail in detecting the characters with the large deformation or distortion. Qiao Y and et al. [13] proposed a technique by determining an equation for calculating the desired threshold which used to separation image into object and background, this equation fundamentally to deal with small objects and relies on upon variance and intensity contrast between object and background.

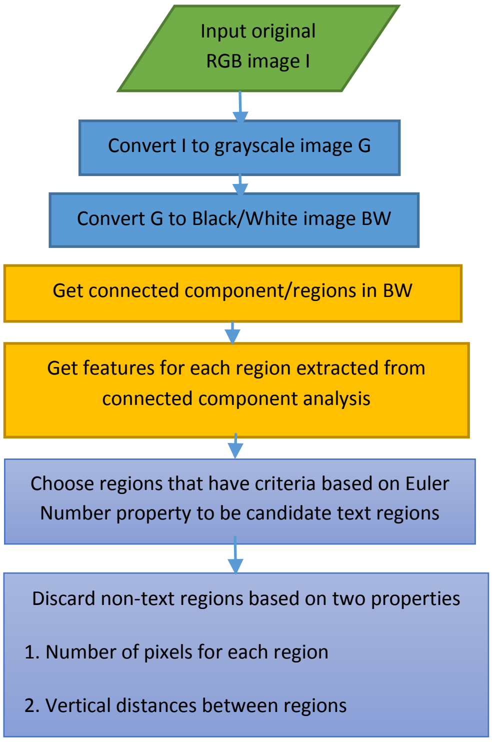
Input original RGB image I
Convert I to grayscale image G
Convert G to Black/White image BW
Get connected component/regions in BW
Get features for each region extracted from connected component analysis
Choose regions that have criteria based on Euler Number property to be candidate text regions
Discard non-text regions based on two properties
1. Number of pixels for each region
2. Vertical distances between regions

Fig. 2: Steps of Proposed Method

4. METHODOLOGY

Preliminary:

There are many shape and geometric features that can be used in text detection, these properties exist in [14].

We trace most of them empirically such as ConvexHull, Eccentricity, Extent, Orientation, Solidity, number of pixels, and EulerNumber to explore which of these properties stronger than others in defining character regions.

We found that EulerNumber property is the most strongest to take the maximum number of character regions among all properties. EulerNumber is defined as the number of objects minus the number of holes in these objects, as shown in Fig 3.

EulerNumber = 1-1 = 0 EulerNumber = 1-2 = -1

Fig. 3: Samples of EulerNumber values

Also, if we have two pixels with Cartesian coordinates; **$P(x_1, y_1)$** and **$P(x_2, y_2)$**, the vertical distance computed from this formula:

Vertical Dist. = $| y_1 – y_2 |$.

We use the Vertical distance to determine the character regions existed in same horizontal line or nearly in the same horizontal line by computing this distance between the centers of regions.

As shown in Fig 2, our proposed scheme depend on exploiting some shape properties and geometric features of image regions like EulerNumber, Centroid and number of pixels of image regions. We summarize the algorithm as follow:

1. Input RGB image.

2. Convert RGB image to Grayscale image, then convert the grayscale image to binary image.

3. Apply the connected components analysis on the binary image to extract the non-zero regions.

4. Extract some shape properties of these regions using the regionprops function.

5. Apply EulerNumber property to define text regions by suitable threshold, we can notice that the minimum holes is zero for some letters so the EulerNumber is 1, and the maximum holes is two holes so the corresponding EulerNumber is -1. Although these values is identical of the EulerNumber for any character we should consider some noise existed in some characters, so we consider EulerNumber for four holes that is its value is -3.

6. After we candidate all text regions depending on EulerNumber, we apply the second stage, in this stage, we discard the non-text regions using number of pixels, then we compute all vertical distances between each two regions, we discard the region has maximum distances between this region and other regions in the image.

5. EXPERIMENTAL RESULTS

We applied our scheme on the same dataset which we used in our previous work. This dataset is gathered from well-known data sets ICDAR2013, ICDAR2015, the datasets contain 59 images with different font size of letters and almost equivalent font size of letters. Because we use connected-component analysis as an initial step in detection, we intend to select the images whose letters are brighter than the background and other elements. Our methods can apply on image whose text is darker than the background by inversing the image. Firstly, we try our scheme using a base feature to get all candidate characters, this base feature is EulerNumber.

Then a single feature is used as a filter to discard non-text regions, this discarding feature is number of pixels for each region. Secondly, we run the scheme by adding another discarding feature, this feature is the vertical distance between each two regions. When we present the results, we divide the results of all set into two sets, the first is images whose text consist almost equivalent size letters, and the second whose text consists of different size letters. We used the vertical distance to decrease the false positive, we apply the two schemes; EulerNPixel and EulerNPixelDist as shown in tables and figures by choosing the threshold of discarding feature once through empirical heuristic rules and again as a fixed threshold (TH = 100 pixels).

These heuristic rules are derived based on the pixels mean value of all regions and quartiles of pixels number of all regions. We can define the quartiles from [15] as follow:

The quartiles of a ranked set of data values are the three points that divide the data set into four equal groups, each group comprising a quarter of the data

The first quartile (Q1) is defined as the middle number between the smallest number and the median of the data set. The second quartile (Q2) is the median of the data. The third quartile (Q3) is the middle value between the median and the highest value of the data set.

This is the heuristic rule for choosing threshold:

```
if  MEAN > 100
    TH =  QUARTILE (1);
elseif  MEAN < QUARTILE (3)
    TH = QUARTILE (2);
else
    TH = 3 * MEAN;
End
```

We compute the basic metrics as follow:
Precession = TP / (TP + FP)
Accuracy = (TP + TN) / (TP + TN + FP + FN)
Recall = TP / (TP + FN)

Table1: Overall + Threshold = 100

	Bayes2	EulerNPixel	EulerNPixelDist
Precission	0.66264	0.85821455	0.869266055
Accuracy	0.560967185	0.92281208	0.923822252
Recall	0.466019	0.69333333	0.689090909

Table2: Equal size letters + Threshold = 100

	Bayes2	EulerNPixel	EulerNPixelDist
Precission	0.602911	0.819148936	0.836244541
Accuracy	0.67570009	0.970512348	0.971976401
Recall	0.633188	0.836956522	0.832608696

Table3: Different size letters+ Threshold = 100

	Bayes2	EulerNPixel	EulerNPixelDist
Precission	0.705015	0.878751501	0.887058824
Accuracy	0.489932886	0.86806161	0.850351617
Recall	0.401681	0.674654378	0.633613445

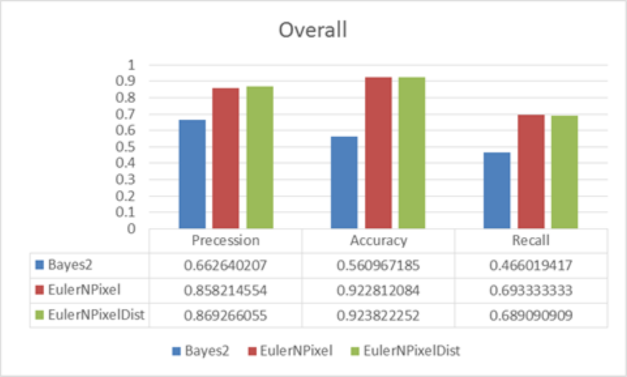

Fig. 4: Comparison between three schemes at all dataset

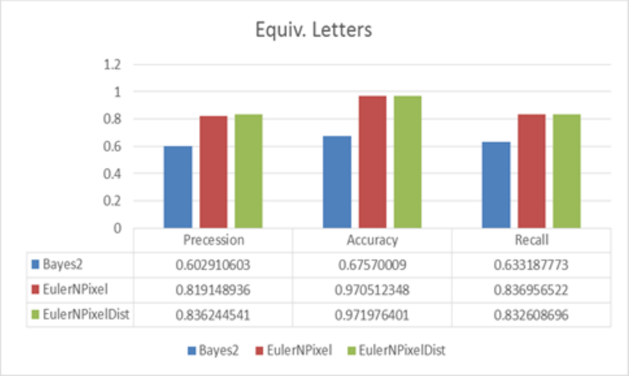

Fig. 5: Comparison between three schemes at equivalent font size letters case

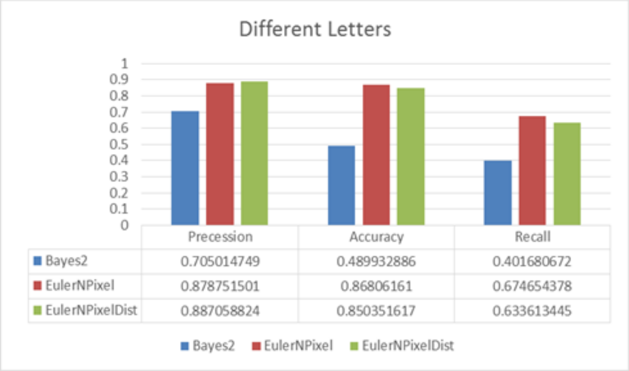

Fig. 6: Comparison between three schemes at at different font size letters case

In the previous tables; Table1, Table2, and Table3 and graphs; Fig4, Fig5, and Fig6, we show that our new methods EulNPixel and EulNPixelsDist give results better than the previous method Bayes2, Also, these methods give the least false alarms in comparison of our previous method Bayes2, When we used the second feature, vertical distance as a filter in EulNPixelDist, the false alarm decreases. Generally, the EulNPixelDist give the best accuracy as shown in the tables and figures but when letters are different the EulNPixel gives the best accuracy.

Now we show in Table4, Table5, and Table6 the difference between two types of choosing threshold, one of them is fixed threshold (we choose TH =100), and the other is dynamic or automatic threshold , this is based on some statistical measures like the mean and the quartiles of the candidate regions pixels.

Table4: Overall

	TH/100	TH/heuristic rules
Precission	0.869266055	0.465793304
Accuracy	0.923822252	0.795388728
Recall	0.689090909	0.776228017

Table5: Equal size letters

	TH/100	TH/heuristic rules
Precission	0.836244541	0.385718134
Accuracy	0.971976401	0.805089434
Recall	0.832608696	0.819172113

Table6: Different size letters

	TH/100	TH/heuristic rules
Precission	0.88705882	0.646638054
Accuracy	0.85035162	0.780590717
Recall	0.63361345	0.759663866

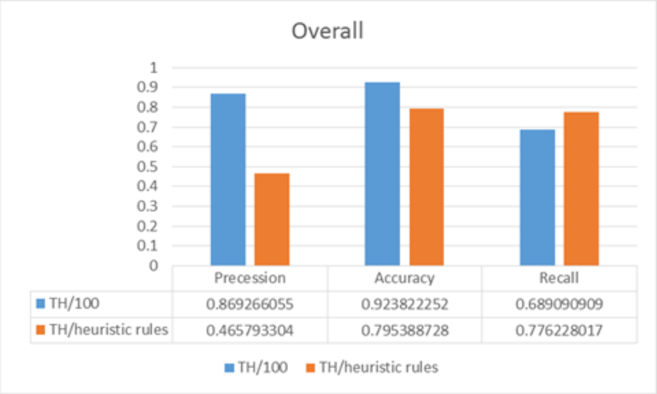

Fig. 7: Comparison between two types of threshold in the third method EulNPixelDist at all dataset

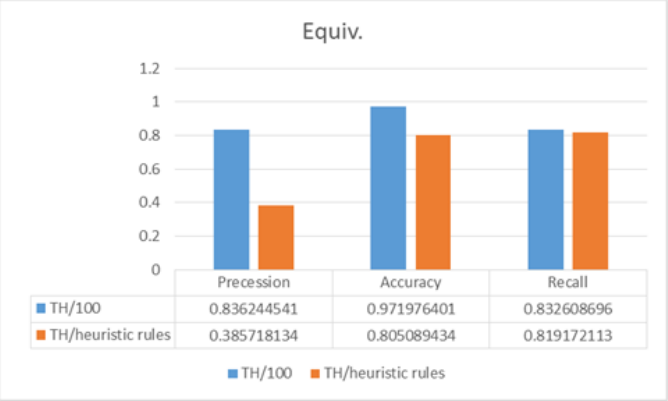

Fig. 8: Comparison between two types of threshold in the third method EulNPixelDist at equivalent font size letters case

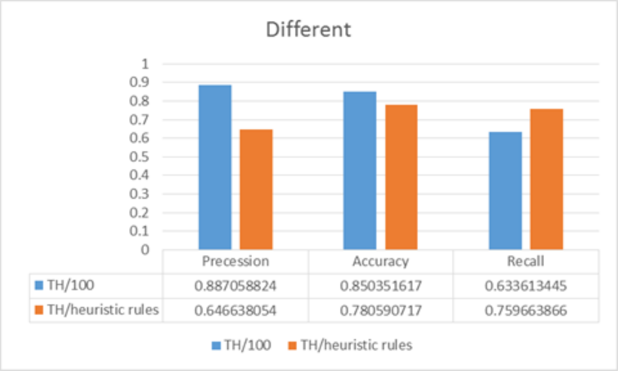

Fig. 9: Comparison between two types of threshold in the third method EulNPixelDist at different font size letters case

It is shown from previous tables and graphs that when we fix the threshold the false alarms is less than when threshold is automatic in our dataset.

This is because of lack of heuristic rules used to define threshold and this is result in the statistical measures used are not suitable for this case , we can benefit from this result that knowing the distribution belongs the number of regions pixels can be participate in defining a good heuristic rules.

Now we present in Fig.10, samples of images which indicate that our new scheme EulNPixelDist results is better than EulNPixel results and the previous one in past work Bayes2.

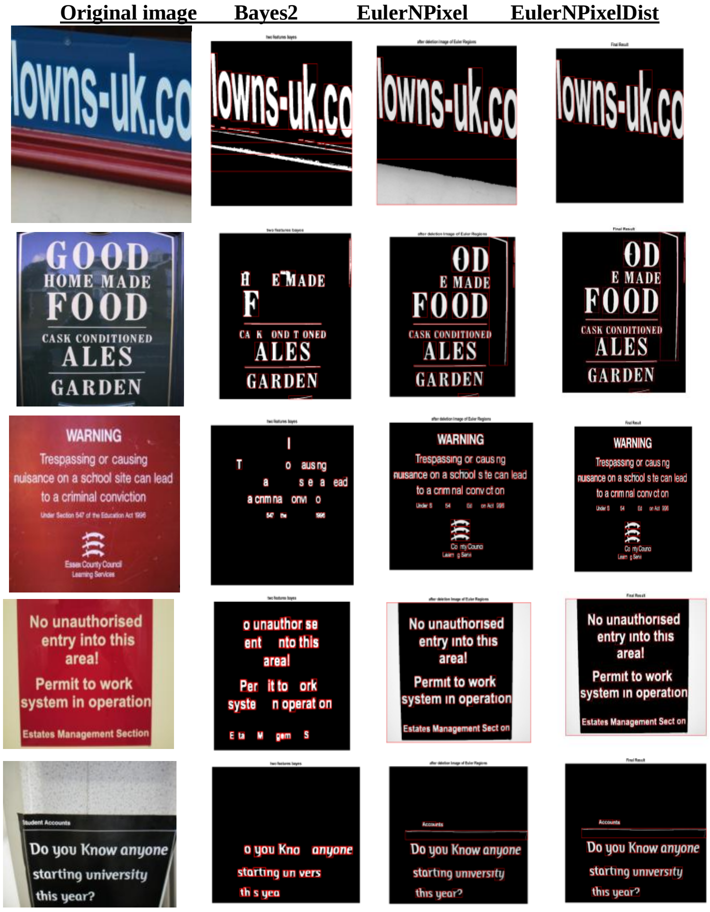

Fig. 10: Samples of results showing the difference between the three methods

Now we present in Fig.11, samples of images which indicate that fixed threshold is better than dynamic one in our dataset.

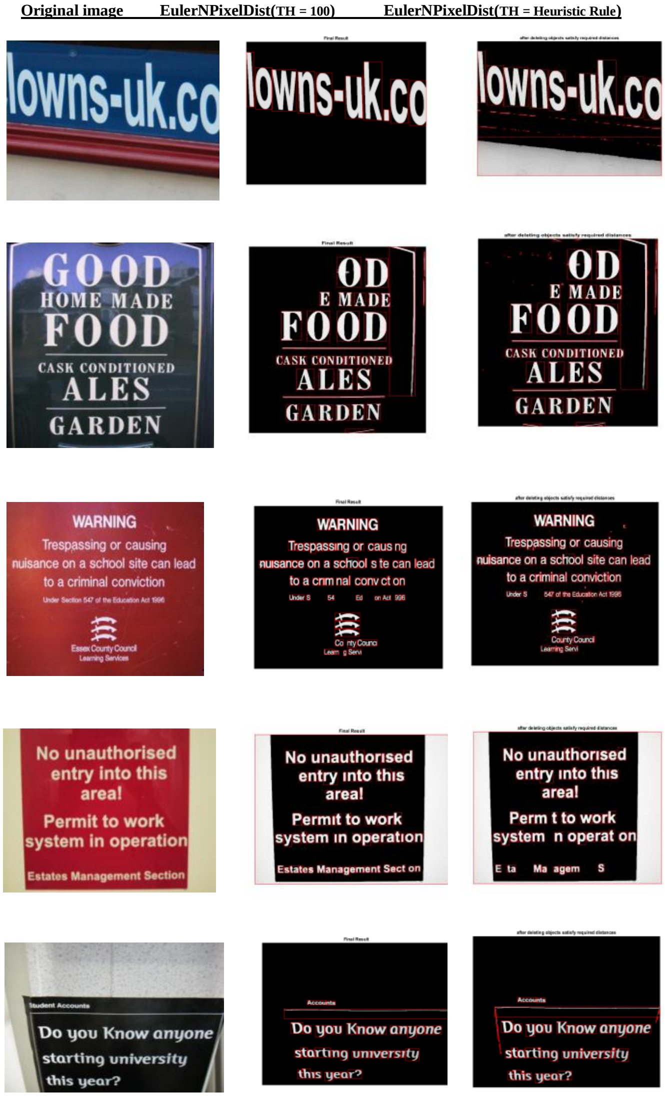
Original image
EulerNPixelDist(TH = 100)
EulerNPixelDist(TH = Heuristic Rule)

Fig. 11: Samples of results showing the difference between the fixed threshold and automatic threshold.

6. DISCUSSION AND CONCLUSION

In this paper, we explore some shape properties and geometric features of image regions. After we test most of this shape properties; ConvexHull, Eccentricity, Extent, Orientation, Solidity, and Eulerumber, we found that the strongest property which tacks the maximum number of text regions is EulerNumber.

So, we take this property as a base property in defining the text regions. Then we use another two features to discard the non-text regions. The first one is number of pixels of each candidate region, and the second features the vertical distances between each two candidate regions, we exclude the area whose distance from the other regions is more distant with a certain threshold value for the distance.

As a future work, more related features will be incorporated to our classifier, including entropy, and Solidity, Also, we try to introduce a new handling for the vertical distances between candidate regions will be more effective in defining and discarding text regions.

7. REFERENCES

[1] Qixiang Ye and David Doermann, Text Detection and Recognition in Imagery: A Survey, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 7, pp. 1480-1500, June 2015.

[2] Chunmei Liu and et al., Text Detection in Images Based on Unsupervised Classification of Edge-based Features, in Proceedings of Eight International Conference on Document Analysis and Recognition, 2005, pp. 610-614

[3] Shivakumara P. and et al., Accurate video text detection through classification of low and high contrast images, Pattern Recognition, 43(2010), 2165-2185.

[4] Jie Y. and et al., A method for text line detection in natural images , Multimed Tools Appl (2015) 74: 859-884.

[5] Kita K. and Toru W., Binarization of Color Characters in scene images Using k-means Clustering and Support Vector Machines, International Conference on Pattern Recognition, 2010.

[6] Honggang Zhang and et al, Text extraction from natural scene image: A survey, Neurocomputing, 122,(2013)

[7] Shivakumara P. and et al., Multioriented Video Scene Text Detection Through Bayesian Classification and Boundary Growing, IEEE Transaction on Circuits and systems for Video Technology, vol. 22, no. 8, 2012.

[8] Shivakumara P. and et al, A new multi-modal approach to bib number/text detection and recognition in Marathon images, Pattern Recognition, 61, pp. 479-491, 2017.

[9] Sue We and Adnan Amin, Automatic Thresholding of Gray-level Using Multi-stage Approach, Proceedings of the Seventh International Conference on Document Analysi Recognition, IEEE, 2003.

[10] Abdel-Rahiem Hashem and et al., A Comparison study on text detection in scene images based on connected component analysis, IJCSIS, vol. 15, no. 2, pp. 127-139, 2017

[11] Matias Valdenegro- Toro and et al, Histogram of Stroke Width for Multi-script Text Detection and Verification in Road Scenes, IFAC , 2017.

[12] Shi C. and et al., End-to-end scene text recognition using tree-structure models, Pattern Recognition, 47, pp. 2853-2866, 2014

[13] Yu Qiao and et al., Thresholding based on variance and intensity contrast, Pattern Recognition, 40, pp. 596-608, 2007

[14] Regionprops. Measure properties of image regions https://www.mathworks.com/help/images/ref/regionprops.html

[15] Quartile. From Wikipedia: https://en.wikipedia.org/wiki/Quartile.