MAPS: Midline analysis and propagation of segmentation

Deepak Kumar*
MILE Laboratory
Dept. of EE
Indian Institute of Science
Bangalore, INDIA
deepak@ee.iisc.ernet.in

M N Anil Prasad
MILE Laboratory
Dept. of EE
Indian Institute of Science
Bangalore, INDIA
anilprasadmn@ee.iisc.ernet.in

A G Ramakrishnan
MILE Laboratory
Dept. of EE
Indian Institute of Science
Bangalore, INDIA
ramkiag@ee.iisc.ernet.in

ABSTRACT

Scenic word images undergo degradations due to motion blur, uneven illumination, shadows and defocussing, which lead to difficulty in segmentation. As a result, the recognition results reported on the scenic word image datasets of ICDAR have been low. We introduce a novel technique, where we choose the middle row of the image as a sub-image and segment it first. Then, the labels from this segmented sub-image are used to propagate labels to other pixels in the image. This approach, which is unique and distinct from the existing methods, results in improved segmentation. Bayesian classification and Max-flow methods have been independently used for label propagation. This midline based approach limits the impact of degradations that happens to the image. The segmented text image is recognized using the trial version of Omnipage OCR. We have tested our method on ICDAR 2003 and ICDAR 2011 datasets. Our word recognition results of 64.5% and 71.6% are better than those of methods in the literature and also methods that competed in the Robust reading competition. Our method makes an implicit assumption that degradation is not present in the middle row.

Keywords

Segmentation, text recognition, Bayesian classification, max-flow, midline, ICDAR 2003 dataset, ICDAR 2011 dataset, optical character recognition, min-max method, word recognition, propagation

1. INTRODUCTION

Segmentation is an active research area, in image processing, for object detection and recognition. In document analysis systems, early research was focused on segmentation of scanned documents, known as binarization and recognition of characters, known as Optical Character recognition (OCR). Both binarization and recognition are required in the process of document digitization. Several good character recognition (OCR) engines are available for Roman script recognition [24, 29, 25, 30]. Also, several algorithms exist for binarizing highly degraded scanned documents.

Figure 1: Samples from ICDAR 2011 word images dataset [20].

Now, the scope of document analysis systems has expanded to encompass the recognition of numerals in vehicle number plates, text in street names and advertisement text in hoarding boards from camera captured images. A document captured using a camera undergoes perspective deformation, page curl, non-uniform illumination and blur. When we pass such a camera captured image directly to an OCR for recognition, the performance is poor. When compared to a scanned document image, camera-captured images require different kind of processing. Hence, text localization, text segmentation and word recognition have become major research areas of camera captured images.

Lucas et.al divided the task of robust reading of text in camera captured images into two parts, namely detection and recognition. Lucas et.al introduced and organized separate competitions in International Conference on Document Analysis and Recognition (ICDAR) 2003 [10] for text localization on camera captured images and recognition of the word images extracted manually by placing a bounding box on those images. They received five entries for text localization and none for word recognition. In continuation, Lucas et.al conducted only text localization competition in ICDAR 2005 [11]. Now, we have several publicly available datasets

*Corresponding author

ICVGIP '12, December 16-19, 2012, Mumbai, India

for robust reading task [26]. These datasets are known as IAPR TC11 Reading Systems-Datasets. Text localization research took the main stage, since it was assumed that passing the bounding box information about a word is sufficient for good recognition of the text by an OCR. But, we observe that the best performing algorithm has only 61% word recognition rate on ICDAR 2003 word image dataset [22], even though they used a customized lexicon derived from the set of test images themselves for word recognition. In a real scenario, depending on such custom lexicons might limit the scope of word recognition, as we cannot predict the text that may appear in a scene. Thus, in our experiments, we avoid using lexicon derived from test images. Recently held ICDAR 2011 Robust Reading challenge 2 reports 41.2% as the best word recognition rate, among the competed methods [20]. In Figure 1, we show sample images from ICDAR 2011 dataset.

The major problem in processing scenic word images is degradation and it is difficult to segment a degraded word image. Also, during the binarization process, several characters get merged due to small character gaps, thus reducing the recognition rate of word dataset. Thus, in this paper, we undertake the task of developing an effective algorithm to binarize word images. On the other hand, character recognition has been extensively researched by document imaging community. So, we use trial version of Omnipage OCR [29] for recognition of characters in the binarized image.

2. RELATED WORK

Here, we discuss some of the methods in the literature to know the issues in the problem of word recognition. At preprocessing stage, TH-OCR system normalizes each word image to a fixed height of 100 pixels using bi-cubic interpolation [20]. In ICDAR 2003 and ICDAR 2011 datasets, we observe variation in the stroke width of the text components. We perform a normality test to select a height range for images rather than fixing the height for the entire dataset. Methods that have been proposed for segmenting word images are conditional random fields (CRFs) to form super pixels by KAIST AIPR [28], Maximally Stable Extremal Regions (MSER) by Neumann [8], [17] and Markov random fields (MRFs) by Mishra et.al [19]. During segmentation, Mishra et.al have used Canny edges to seed foreground and background pixels [3]. Other methods that have been explored are clustering and combining different segmentation techniques [15, 16, 18, 13, 14].

After binarization of word images, recognition is performed using either a standard OCR engine or a classifier built for the purpose. KAIST AIPR system classifies super pixels and passes them to INZI soft OCR engine [27]. Similarly, TH-OCR system uses an OCR engine [12], Mishra et.al use ABBYY OCR reader [24], Zeng et.al use OmniPage OCR reader [29] and Neumann et.al classify detected characters in the image using multi-class support vector machines (SVM) [17]. Neumann et.al use character contour feature for classification. Due to variation in the contour feature caused by noise or scaling, they were ranked low in the ICDAR 2011: Robust Reading Competition [20]. This indicates that improvement is required for the features in the classification stage.

Word recognition can also be performed in one go (single stage), using a training dataset, which avoids binarization. Wang et.al [21] and Mishra et.al [22] follow such a strategy using a lexicon in their top-down approach. Since ICDAR 2003 dataset does not provide any lexicon for text recognition, Wang et.al have created a custom lexicon for this dataset, which is available as a part of the Street View Text (SVT) dataset [21]. Both Wang et.al [21] and Mishra et.al [22] make use of this custom lexicon for text recognition. Since custom lexicon prevents the possibility of recognition of new sets of images, our method does not use any lexicon to improve the recognition rate.

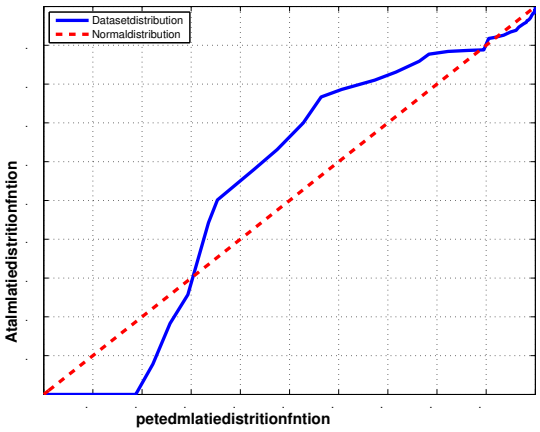

Figure 2: Q-Q plots on image heights for ICDAR 2011 dataset. The actual cumulative distribution function is plotted against the expected cumulative distribution, before height normalization of images.

3. MIDDLE ROW METHOD

In this section, we propose a unique method which picks the middle row of the word image for segmentation. Then for binarizing the entire image, the label information of middle row pixels is passed on to other pixels. Certain preprocessing and post-processing steps are required to standardize the size of images in the dataset.

3.1 Height normalization

We do not have information about the stroke width of characters in the dataset. Apriori, we have access only to the height and width of the word images. A normality test was conducted on image heights in the dataset [7]. In Figure 2, we show the plot of actual cumulative distribution against expected cumulative distribution. We observe that the image heights in the dataset are not close to the diagonal plotted in Figure 2. So, we perform image scaling to normalize the heights of the image in the dataset. Images are scaled by bi-cubic interpolation preserving the aspect ratio. In order to minimize the variance in the stroke width, we modify the height to lie within a range. A height range of 60 to 180 pixels was obtained by performing normality test. Accordingly the rules for rescaling are:

- Rule 1: If the height of an image is less than 60 pixels, then it is rescaled by a factor of '3'.
- Rule 2: If the height of an image lies between 60 and 180 pixels, then it is not rescaled.
- Rule 3: If the height of an image exceeds 180 pixels, then it is scaled down to a height of '180' pixels.

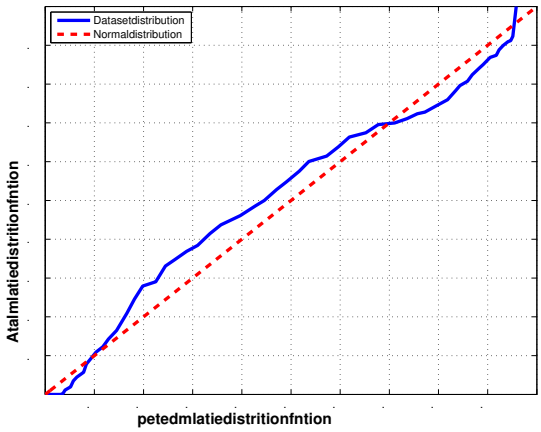

Figure 3: Q-Q plots on image heights for ICDAR 2011 dataset, after height normalization of images. The actual cumulative distribution function is plotted against the expected cumulative distribution.

After height scaling based on these rules. We again show the plot of actual cumulative distribution against expected cumulative distribution in Figure 3. We observe the plot to be approximately linear, which results in a close to normally distributed dataset.

For ICDAR 2003 dataset also, we show the plots of actual cumulative distribution against expected cumulative distribution in Figures 4 and 5, before and after height scaling, respectively.

3.2 Segmentation of Mid-line

We select the middle row of the image and segment it independently using, Niblack and Min-Max Methods. The purpose behind two separate segmentations is to examine any variation in recognition rate, based on computationally simple and expensive methods.

3.2.1 Niblack Method

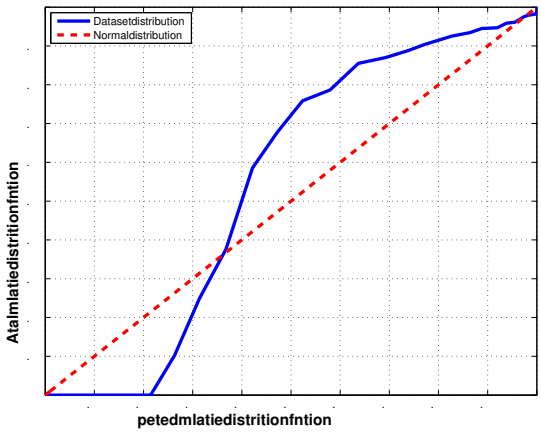

Figure 4: Q-Q plots on image heights for ICDAR 2003 dataset. The actual cumulative distribution function is plotted against the expected cumulative distribution, before height normalization of images.

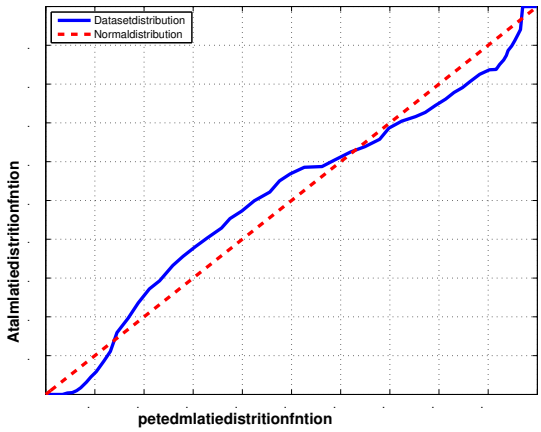

Figure 5: Q-Q plots on image heights for ICDAR 2003 dataset, after height normalization of images. The actual cumulative distribution function is plotted against the expected cumulative distribution.

Niblack proposed an algorithm to calculate a local threshold for each pixel by moving a rectangular window over the whole image [4]. In our method, we modified the Niblack method to one dimension. The mean and the standard deviation values of all the pixels in the window are used to calculate the threshold. Thus, the threshold is given as:

$$T_i = \mu_i + \ k_n \ * \ \sigma_i \tag{1}$$

$$\mu_i = \frac{1}{N_w} \sum_{j \in N_w} x_{i+j} \tag{2}$$

$$\sigma_i = \frac{1}{N_w} \sum_{j \in N_w} (x_{i+j} \ - \ \mu_i)^2 \tag{3}$$

Here, 'x_i' is the gray value at pixel position i in the middle row. User has to fix the values for 'k_n' and window size. We have used $k_n = 0.1$ and $N_w = min(height, width)$. In our experiment, these values are fixed for all the word images tested.

3.2.2 Min-Max Method

Methods that are locally adaptive make use of local statistics to infer the presence or absence of the two classes within the window. In the case of bimodal distributions, the values corresponding to Min and Max belong to different distributions. This Min and Max values are not dependent on the representation of each distribution within the window. But in the presence of noise, these Min and Max filters get heavily biased and we cannot infer either Min or Max value. By using a Min-Max filter with a carefully chosen size of window, it is possible to mitigate the effect of noise. We obtain the maximum values in the windows placed to the left and right-side of position i and perform minimum operation to obtain T_{max}. Similarly, maximum operation is performed on the minimum values in the left and right-side windows to obtain T_{min} .

$$T_{max} = \min(\max_{N_m} L(x_i), \max_{N_m} R(x_i)) \tag{4}$$

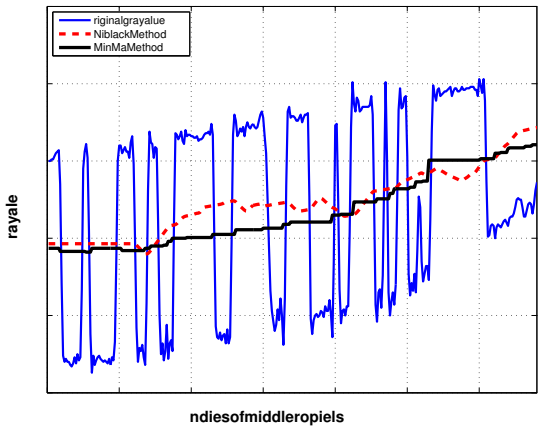

Figure 6: A plot of gray values of the middle row of an image with the binarization thresholds given by Niblack and Min-Max Methods.

$$T_{min} = \max(\min_{N_m} L(x_i), \min_{N_m} R(x_i)) \tag{5}$$

$$L(x_i) = [x_{i-N_m+1}, ..., x_i] \tag{6}$$

$$R(x_i) = [x_i, ..., x_{i+N_m-1}] \tag{7}$$

$$T = (T_{min} + T_{max})/2 \tag{8}$$

where, 'N_m' is the size of the moving window obtained as $N_m = min(height, width)$. The window size was fixed in our experiment, since varying the size did not improve the segmentation.

In Figure 6, we show the plot of gray values of the middle row from a degraded image. The local thresholds obtained by Niblack and Min-Max methods are also plotted. We observe that the variation in the Niblack method is more than that of the Min-Max method. Niblack method calculates the threshold by averaging in a window. When the window slides, this value varies. But in Min-Max method, when we slide a window, the variation of maximum and minimum values is less. Hence, the threshold of Min-Max method is more stable than that of Niblack method.

3.3 Classification of other pixels

The labels obtained from middle row segmentation are used to estimate the means and variances of the two classes.

$$\mu_0 = \frac{1}{N_0} \sum_{i \in C_0} x_i \tag{9}$$

$$\sigma_0^2 = \frac{1}{N_0} \sum_{i \in C_0} (x_i - \mu_0)^2 \tag{10}$$

where μ_0 is the mean of class C_0, which has N_0 labels. σ_0^2 is the variance of class C_0. Similarly, we calculate the mean μ_1 and variance σ_1^2 for class C_1.

Only middle row is considered for segmentation and parameter estimation. Other pixels need to be labeled through classification. We use two different approaches for classification namely, Bayesian classification and Min-Cut/Max-Flow algorithm. Here, we have proposed computationally simple and expensive methods to study their impact on the recognition rate. The worst case running time complexity for Bayesian classification is O(N), whereas for Min-Cut/Max-Flow algorithm, it is O(N^3). Here, N is the number of pixels in the image.

3.3.1 Bayesian classification

In Bayes binary classification, the posterior probability of sample $x_{i,j}$ belonging to class C_0 is given as [6]:

$$p(C_0|x_{i,j}) = \frac{p(x_{i,j}|C_0)p(C_0)}{p(x_{i,j}|C_0)p(C_0) + p(x_{i,j}|C_1)p(C_1)} \tag{11}$$

where $x_{i,j}$ is the gray value at pixel position (i, j). The prior probability is

$$p(C_0) = N_0/N \tag{12}$$

For classification,

$$h(x_{i,j}) : \begin{cases} p(C_0|x_{i,j}) \geq p(C_1|x_{i,j}), & x_{i,j} \in C_0 \\ p(C_0|x_{i,j}) < p(C_1|x_{i,j}), & x_{i,j} \in C_1 \end{cases} \tag{13}$$

The class-conditional density is assumed as Gaussian for binary classification [6]. It is shown below as:

$$p(x_{i,j}|C_0) = \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp\{-\frac{(x_{i,j} - \mu_0)^2}{2\sigma_0^2}\} \tag{14}$$

Using Equation (13), other pixels in the image are classified. Then, the classified pixels representing a binarized word image, are passed to the polarity inversion module.

3.3.2 Min-cut/Max-flow algorithm

In Bayesian method, individual pixels are considered at the time of classification and neighborhood pixel gray values are ignored. To add neighborhood gray values as smoothness term at classification stage, we use graph cut, which was proposed for image segmentation. Pixels are represented as graph nodes with edges connected to other pixels. The energy function of the Potts model [9], which has to be minimized is given as

$$E(L) = \sum_{i \in \mathcal{I}} D_i(L_i) + \sum_{(i,j) \in \mathcal{N}} V_{i,j}(L_i, L_j) \tag{15}$$

where $L = L_i | i \in \mathcal{I}$ is a labeling of the image $\mathcal{I}$, $D(.)$ is a data penalty function, $V_{i,j}$ is an interaction potential, and $\mathcal{N}$ is a set of all pairs of neighboring pixels.

We perform energy minimization by incorporating the means and variances as parameters into data penalty and interaction potential functions in Boykov et.al program [9]. The minimum energy results in a binarized image, which is sent to the polarity inversion module.

In Figure 7, we show the result for a sample image drawn from ICDAR 2003 dataset. The results are shown using Otsu's [1], Canny's [3], Niblack's [4] and MAPS method. From the results, it is evident that even degradations like uneven illumination and low contrast are effectively handled by our method.

Figure 7: Segmentation of a sample word image from ICDAR 2003 dataset. (a) Original image. (b) Output of Otsu's method [1]. (c) Edges by Canny's method [3]. (d) Binarized output of Niblack's method [4]. (e) Output of MAPS technique using Min-Max method for segmentation and Max-flow algorithm for classification.

3.4 Polarity inversion of text as needed

Figure 8 shows two sample binarized images. Here, black and white pixels denote background(0) and foreground(1) respectively. Such binarized images result in wrong recognition when passed through standard OCR engines, since both foreground and background pixels touch the image boundary. Hence, we propose optional text polarity inversion and a background padding stage after inversion step to remove the foreground-background ambiguity.

Text polarity is detected by examining the following three conditions:

- Is the ratio of number of white pixels along the boundary of segmented word image to the length of boundary greater than 0.5?
- Is the ratio of number of white pixels on the vertical sides of the segmented word image to the total length of the side walls greater than 0.5?
- Is the ratio of maximum widths of 'white' to 'black' connected components in the segmented word image greater than 1?

If two out of these three conditions are true, then polarity needs to be inverted, after which the text pixels will be white.

During thresholding stage, uneven illumination causes salt and pepper noise in the binarized image. Hence, we perform median filtering with a structuring element of size 5x5. The images rescaled using Rule 1 are excluded from median filtering, since they are of low resolution and filtering may degrade the binarized image.

To prevent the text connected components from touching the boundary of the image, we pad zeros, vertically by half the number of rows and horizontally by half the number of columns. An example is shown in Figure 9. In Figure 9(i) the text connected components touch the word boundary. After padding zeros in the image, we clearly observe the distinction between foreground and background pixels. Post-processed binarized word image is fed into the recognition engine.

Figure 8: The background is broken, where the text connected components touch the boundary of the word image. This creates ambiguity in determining the text polarity.

(i) The text connected components touch the boundary

(ii) Foreground and background are clearly separated

Figure 9: Binarized image before and after padding zeros.

4. RECOGNITION OF WORD

We pass the binarized word image to an OCR engine. There are several open access and professional OCR engines such as Tesseract [30], OmniPage [29], Adobe Reader [25] and Abbyy Fine Reader [24]. The main application of OCR engines is in the digitization of old hard bound documents. The recognition performance of OCR engines is good on clean document images. We use the trial version of Omnipage OCR engine for recognition of the word images. The number of words correctly recognized is noted to evaluate the effectiveness of MAPS algorithm.

5. EXPERIMENTAL RESULTS

ICDAR 2003 word images training dataset contains 1156 images extracted from scenic images. Testing dataset consists of 1110 word images. Whereas, ICDAR 2011 dataset has 849 and 716 word images for training and testing, respectively. We have applied our binarization method on each test image and recognized the word. During the experiments, we observe that, there exists some correlation between the images used in both datasets. Apart from correlation, boundaries are improper in few cropped images of ICDAR 2003 dataset. The cropped images in ICDAR 2011

Table 1: Performance evaluation of MAPS algorithm on ICDAR 2003 dataset for different choices of segmentation and label propagation methods.

Segmentation methods + Propagation methods for MAPS	Word recognition rate (%)	Character recognition rate (%)
Niblack + Bayes	63.2	78.8
Niblack + Max-flow	63.6	79.0
Min-Max + Bayes	64.7	78.7
Min-Max + Max-flow	**64.5**	**79.2**

Table 2: Performance evaluation of MAPS algorithm on ICDAR 2011 dataset for different choices of segmentation and label propagation methods.

Segmentation methods + Propagation methods for MAPS	Edit distance	Word recognition rate (%)
Niblack + Bayes	192.7	70.7
Niblack + Max-flow	198.2	70.1
Min-Max + Bayes	201.6	71.4
Min-Max + Max-flow	**199.7**	**71.6**

dataset are tight or closely bounded, which sometimes cause the characters to touch the boundary.

Table 1 shows word and character recognition rates for ICDAR 2003 dataset with different variations proposed in MAPS algorithm. Table 2 shows edit distance and word recognition rate for ICDAR 2011 dataset. Edit distance measure was introduced as one of the performance measures in ICDAR 2011 competition. Equal weights are given for additions, substitutions and deletions. Normalized edit distance is calculated between the transcription of ground-truth and output of MAPS algorithm. Even though we have applied two extremes of computational methods, the variation in the observed results with different varieties of MAPS algorithm is very less and the results differ by a maximum of 1% as reported in Tables 1 and 2.

Table 3 reports the word recognition rate of our method on ICDAR 2003 database and compares it with the results in the literature. During performance evaluation on ICDAR 2003 dataset, some of the algorithms in the literature have ignored low resolution and degraded images. We have averaged the results of those algorithms over the entire dataset for use in Table 3. Table 4 shows the edit distance measures and word recognition rates of our method as well as others on ICDAR 2011 dataset. The edit distance of our method is higher on ICDAR 2011 dataset, due to misclassification of pixels resulting in noise near the boundary of the image.

We have used the best results of our algorithm from Tables 1 and 2 for comparison with the results of other algorithms in Tables 3 and 4. However, it may be noted that the comparison and the relative trends will be identical, even if we had used the worst of the results from those tables, since, as we have already mentioned, the different results don't differ by more than 1% between one another.

6. DISCUSSION

Each of our processing steps plays an important role in improving the recognition rate on ICDAR datasets. From Tables 3 and 4, we can observe that the performance of MAPS method is better than others on ICDAR 2003 dataset and far exceeds those of others on ICDAR 2011 dataset. The major factor we have handled in the algorithm is illumination variation during the binarization stage. This ability is achieved by picking up the middle row as the sub-image for segmentation. Thus, the improvement in segmentation provides boost in improving the word recognition rates. To verify the validity of our assumption of minimal degradation in the middle line of an image, we have compared our middle line segmentation results with the ground truth middle line for each dataset [23] and found that nearly 90% of the images in the dataset match with our segmentation methods.

Table 3: Comparison of performance of MAPS algorithm with those of algorithms in the literature on ICDAR 2003 dataset.

Algorithm	Word recognition rate (%)	Character recognition rate (%)
MAPS algorithm	**64.5**	**79.2**
Mishra et.al [22]	61.1	—
Wang et.al [21]	53.8	—
Zeng et.al [18]	—	75.3
Otsu [1]	38.4	56.8
Kittler et.al [2]	37.8	55.2
Sauvola [5]	22.5	39.1
Niblack [4]	18.7	38.0

Table 4: Comparison of performance of MAPS algorithm with those of algorithms in the literature on ICDAR 2011 dataset.

Algorithm	Edit distance	Word recognition rate (%)
MAPS algorithm	**199.7**	**71.6**
TH-OCR System	176.4	41.2
KAIST AIPR System	318.5	35.6
Neumann's Method	429.7	33.1
Otsu [1]	596.4	18.2
Kittler et.al [2]	644.6	18.0
Sauvola [5]	763.5	15.9
Niblack [4]	1469.4	12.7

We have used image statistics itself, while classifying the pixels in the image. Hence, this method resembles a bottom-up approach.

Figure 10 shows two sample word images, where the proposed algorithm fails to recognize the word correctly. Strong illumination and low contrast appear in the middle row, which violate our assumption, resulting in failure of the algorithm in those images.

Figure 10: Word images, where proposed algorithm fails to recognize the words contained in them.

7. CONCLUSION AND FUTURE WORK

We have proposed an algorithm for effective segmentation of words from different word image datasets. We observe from Tables 3 and 4 that we cannot extract all the words perfectly. This could be due to artistic font, severe degradations and/or varying stroke width in the word image dataset. In our future work, we plan to use multiple rows at the sub-image segmentation stage to avoid dependency on middle row, include colour information at classification stage and estimate accurate stroke width to reduce the effect of degradations caused to text pixels.

8. REFERENCES

[1] N. Otsu, A Thresholding Selection Method from Gray-level Histogram, *IEEE Transanctions on Systems, Man and Cybernetics*, vol. 9, pp. 62–66, March 1979.

[2] J. Kittler, J. Illingworth, and J. Foglein, Threshold selection based on a simple image statistic, *Computer Vision, Graphics, and Image Processing*, vol. 30, no. 2, pp. 125–147, 1985.

[3] J. Canny, A Computational Approach to Edge Detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, November 1986.

[4] W. Niblack, *An introduction to digital image processing*. New York: Prentice Hall, 1986.

[5] J. J. Sauvola and M. Pietäikinen, Adaptive document image binarization, *Pattern Recognition*, vol. 33, no. 2, pp. 225–236, 2000.

[6] R. O. Duda, P. E. Hart and D. G. Stork., *Pattern classification*, Wiley, 2001.

[7] H. C. Thode, Jr., Testing for Normality, New York, Marcel Dekker, 2002.

[8] J. Matas, O. Chum, M. Urban and T. Pajdla, Robust wide baseline stereo from maximally stable extremal regions, *British Machine Vision Conference*, pp. 384–393, 2002.

[9] Y. Boykov and V. Kolmogorov, An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision, *IEEE Trans. PAMI*, vol. 26, no. 9, pp. 1124-1137, September 2004.

[10] S. M. Lucas et.al, ICDAR 2003 Robust Reading Competitions: Entries, Results, and Future Directions, *International Journal on Document Analysis and Recognition*, vol. 7, no. 2, pp. 105-122, June 2005.

[11] S. M. Lucas, Text Locating Competition Results, Proc. 8th *Intl. Conf. on Document Analysis and Recognition (ICDAR) 2005*, pp. 80–85, 2005.

[12] H. Liu, X. Ding, Handwritten Character Recognition Using Gradient Feature and Quadratic Classifier with Multiple Discrimination Schemes, Proc. 8th *Int. Conf. on Document Analysis And Recognition*, pp.19–25, 2005.

[13] C. M. Thillou and B. Gosselin, Color text extraction from camera-captured images: the impact of the choice of the clustering distance, Proc. of 8th *Int. Conf. on Document Analysis And Recognition*, pp. 312–316, 2005.

[14] C. M. Thillou and B. Gosselin, Color text extraction with selective metric-based clustering, *Computer, Vision and Image Understanding*, vol. 107, no. 2, pp. 97–107, 2007.

[15] T. Kasar, J. Kumar and A. G. Ramakrishnan, Font and background color independent text binarization, Proc 2nd *Camera-based Document Analysis and Recognition (CBDAR)*, pp. 3–9, 2007.

[16] T. Kasar and A. G. Ramakrishnan, COCOCLUST: Contour-based color clustering for robust binarization of colored text, Proc 3rd *Camera-based Document Analysis and Recognition (CBDAR)*, pp. 11–17, 2009.

[17] L. Neumann and J. Matas, A Method for Text Localization and Recognition in Real-World Images, Proc. 10th *Asian Conference on Computer Vision (ACCV)*, pp.770-783, 2010.

[18] C. Zeng, W. Jia and X. He, An Algorithm for Colour-based Natural Scene Text Segmentation, Proc 4th *Camera-based Document Analysis and Recognition (CBDAR)*, pp. 67–72, 2011.

[19] A. Mishra, K. Alahari and C. V. Jawahar, An MRF Model for Binarization of Natural Scene Text, Proc. 11th *International Conference of Document Analysis and Recognition*, pp. 11–16 , September 2011.

[20] A. Shahab, F. Shafait and A. Dengel, ICDAR 2011 Robust Reading Competition - Challenge 2: Reading Text in Scene Images, In Proc. 11th *International Conference of Document Analysis and Recognition*, pp. 1491–1496 , September 2011.

[21] K. Wang, B. Babenko and S. Belongie, End-to-End Scene Text Recognition, Proc. 13th *International Conference on Computer Vision (ICCV)*, pp. 1457-1464, 2011.

[22] A. Mishra, K. Alahari and C. V. Jawahar, Top-Down and Bottom-Up Cues for Scene Text Recognition, Proc. *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[23] D. Kumar, M. N. Anil Prasad and A. G. Ramkrishnan, Benchmarking recognition results on word image datasets, *CoRR*, vol. abs/1208.6137, 2012. http://arxiv.org/abs/1208.6137

[24] "Abbyy Fine reader," http://www.abbyy.com/.

[25] "Adobe Reader," http://www.adobe.com/products/acrobatpro/scanning-ocr-to-pdf.html.

[26] IAPR TC11 Reading Systems-Datasets List, http://www.iapr-tc11/mediawiki/index.php/Datasets

[27] "Inzisoft," http://www.inzisoft.com/english/.

[28] "KAIST AIPR," http://ai.kaist.ac.kr/home/.

[29] "Nuance Omnipage reader," http://www.nuance.com/.

[30] "Tesseract OCR engine," http://code.google.com/p/tesseract-ocr/.