Robust Text Segmentation in Low Quality Images via Adaptive Stroke Width Estimation and Stroke based Superpixel Grouping

Anna Zhu, Guoyou Wang, Yangbo Dong

State Key Lab for Multispectral Information Processing Technology,
School of Automation, Huazhong University of Science and Technology, China

Abstract. Text segmentation is an important step in the process of character recognition. In literature, there have been numerous methods that work very well in practical applications. However, when an image includes strong noise or surface reflection distraction, accurate text segmentation still faces many challenges. Observing that the stroke width of text is stable and significantly different from that of reflective regions generally, we present a novel method for text segmentation using adaptive stroke width estimation and simple linear iterative clustering superpixel (SLIC-superpixel) region growing in this paper. It consists of four following steps: The first is to normalize image intensity to overcome the influence of gray changes. The second utilizes the intensity consistency to compute normalized stroke width (NSW) map. The third is to estimate the optimal stroke width through searching for the peak value of the histogram of normalized stroke width, the text polarity is also determined. Finally, we propose a local region growing method for text extraction using SLIC-superpixel. Unlike current existing methods of computing stroke width, such as gray level jump on a horizontal scan line and gradient-based SWT methods, the proposed method is based on the statistics of stroke width in the whole image. Hence the stroke width estimation is not only invariant in scale and rotation, but also more robust to surface reflection and noise than that of those methods based only on the pairs of sudden changes of intensity or gradient maps. Experiments with many real images, such as laser marking detonator codes, notice signatures and vehicle license plates, *etc.*, have shown that the proposed algorithm can work well in noised images and also achieve comparable performance with current state-of-the-art method on text segmentation from low quality images.

1 Introduction

The OCR systems update rapidly nowadays and most of current commercial OCR engines work well on high quality documents. However, the recognition precision is greatly influenced by integrity of the segmented characters. OCR systems perform poorly on noisy and poor quality documents due to the lack of valid image segmentation under various variations, such as non-uniform illumination, surface reflection, perspective distortions, and background outliers,

etc. As a part of text analysis in practical applications, text segmentation in low quality images, also named text binarization, still remains to be an important research topic.

Up to now, there already has been a lot of research on text segmentation reported in the literature and can be mainly classified into two categories: statistical thresholding methods and machine learning methods. In the first category, it has two subclasses: global and local. The global ones employ information on the whole of image. The well known global method is presented by Otsu [1] which is to acquire the optimal segmenting threshold of an entire image by the between-class variance maximization. However, when applied to an image where the distraction has similar grayscale to the text, it will produce bad results due to the inseparability in gray. The local thresholding methods use features only in a locally region and the locally segmented results constitute the final segment result. Among locally adaptive methods, Niblack's method [2] has been proved to give the best performance in those images of spatial grayscale changing slowly. However, it might fail when the text is embedded in complex backgrounds or text polarity is initially unknown. [3, 4] presented some modified versions of the Niblack's method to improve binarized results. Satish's edge based local method [5] uses the gray value of edge pixels to decide the threshold in a window and performs well in license plate recognition (LPR) application, but they still fail to distinguishing text and distraction with similar intensity. The same defect is shared with the current used Maximally Stable Extremal Regions (MSER) method [6] which is often applied to text detection and achieves great success.

Recently, various machine learning methods have been popularly proposed in segmentation [7–9]. The basic idea of machine learning is to extract accurate invariant descriptions for specific target recognition by some kind of pattern clustering or classification algorithm. However, these methods also face different problems, like time consuming, insensitive to affine intensity changes, parameter selection, etc.

As stroke width of texts is not only stable but also significantly different from that of reflective regions, the stroke width can be used to detect text by grouping neighboring pixels with similar stroke width into connected components as letter candidates. The algorithm described in [10] scans an image horizontally, looking for pairs of sudden changes of intensity. Then the regions between changes of intensity are examined for color constancy and stroke width. Surviving regions are grouped within a vertical window of size w and if enough regions are found, a stroke can be presented. The limitations of this method include a number of parameters, such as finding vertical window size w tuned to the scale of the text, inability to deal with arbitrary directions of strokes. Moreover, the method [11] also uses the idea of stroke width similarity for detecting text overlays in video sequences. The limitations of the method include the need for integration over scales and orientations of the filter, and the inherent attenuation to horizontal texts again. Another definition of stroke relates to remote sensing images [12]. In this case, the algorithm defines a stroke as a contiguous part of an image that

forms a band of a nearly constant width and describes a local image operator (called as stroke width transform-SWT) that computes the width of the most likely stroke pixels by using gradient maps. This method has been evaluated on natural scene text images with variant fonts and languages and achieves a desirable performance. However, it is sensitive to noise and changing intensity due to the usage of parallel edges.

Unlike the methods mentioned above based on finding pairs of sudden changes of intensity or pairs of edge pixels with roughly opposite directions, we propose a robust algorithm that extracts normalized stroke width (NSW) based on the image intensity similarity in a surrounding window for text extraction in low contrast images. Then a local region growing method based on superpixel level is applied on the NSW map to recall missed text regions using intensity similarity. This paper focuses only on how text can be separated accurately so as to increase recognition results after the text-line is located. This method is insensitive to noise and changing intensity due to its locally statistical intensity similarity.Experimental results on real images captured by camera, such as detonator laser marking code, notice signal and vehicle license plate, show that the proposed algorithm can perform more robust in poor quality images than the existing methods.

The rest of the paper is structured as follows. Firstly, we introduce the proposed method in Section II. In Section III, we analyze the parameters.Experiments and results are presented in Section IV and conclusions are drawn in Section V.

2 The proposed algorithm

Since human visual system needs only the brightness channel to recognize character shapes [13], we firstly convert the image from color to gray scale, then to further split the text from the gray image.

2.1 Image intensity normalization

In many text images, the text is often obscured due to local shadows or surface reflections. To remove this disturbance, we apply to each pixel of the original image $I(x, y)$ a contrast normalization procedure $I(x, y) \leftarrow 0.5+(I(x, y)\text{-}m)/(3\sigma)$. m and σ are the local mean and standard deviation computed by a doubly binomial weight window of width $2H+1$, where H is the height of the text image.

2.2 Pixel-based stroke width estimation

Considering that most of texts are composed of many lines with constant width, so the stroke width is a significant feature of the text. Inspired by [14] for line detection, we define the intensity consistency. As shown in Fig. 1, $P(x, y)$ locates in a surrounding window with size of $r \times r$ centered in pixel $C(x_o, y_o)$. Then the

similarity between $P(x, y)$ and $C(x_o, y_o)$ can be described as

$$\mathrm{con}(x, y, x_o, y_o) = \begin{cases} 1 & \text{if } |I(x, y) - I(x_o, y_o)| \leq t \\ 0 & \text{if } |I(x, y) - I(x_o, y_o)| > t \end{cases} \tag{1}$$

Where t is the threshold of intensity consistency comparison, $I(x, y)$ and $I(x_o, y_o)$ is intensity of the pixel C and P, respectively.

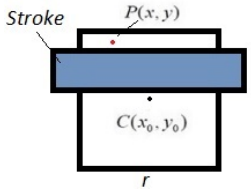

Fig. 1. The surrounding window.

Based on the intensity consistency between center pixel and other pixels in the window, the normalized stroke width (NSW) is calculated as:

$$w(x_o, y_o) = \frac{\sum_{(x,y)\in W_{r\times r}} \mathrm{con}(x, y, x_o, y_o)}{r^2} \tag{2}$$

Where r^2 is the number of pixels in the window, the stroke width of each pixel can be calculated by $w(x_o, y_o, t) \times r$. In order to detect the stroke structure of text, we will discuss the NSW as follows:

(1) If $C(x_o, y_o)$ locates in an uniform background or wider stroke with width exceeding r, $w(x_o, y_o) = 1$.

(2) When $C(x_o, y_o)$ moves onto the boundary of backgrounds adjacent to stroke, if stroke width is larger than half of r, $w(x_o, y_o) = 0.5$, else $w(x_o, y_o) > 0.5$.

(3) When $C(x_o, y_o)$ moves on the boundary of stroke adjacent to backgrounds, if stroke width is larger than half of r, $w(x_o, y_o) = 0.5$, else $w(x_o, y_o) < 0.5$.

(4) When $C(x_o, y_o)$ moves to other locations on the stroke, if stroke width is larger than half of r, $w(x_o, y_o) > 0.5$, else $w(x_o, y_o) < 0.5$.

From the above mentioned, we can see that when $C(x_o, y_o)$ belongs to backgrounds, $w(x_o, y_o) \geq 0.5$. On the contrary, when $C(x_o, y_o)$ locate on stroke, if the stroke width is less than half of r, $w(x_o, y_o) \leq 0.5$, else $w(x_o, y_o) \geq 0.5$. Hence we set the maximum width $w_{max} = 0.5$, the stroke widths are computed by

$$S(x_o, y_o) = \begin{cases} w(x_o, y_o) & \text{if } w(x_o, y_o) < w_{\max} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Where $S(x_o, y_o)$ is normalized stroke image, in which the width is less than 0.5. If we set w_{max} equal to different ranges, those width-specified lines can also be captured. The threshold t and window size r will be discussed in next section. The NSW map is presented in Fig. 2 (b) where the non-zero pixels' values are binary to 1 for show.

In order to estimate the stroke width of text in an image that includes strong noise and reflection interference, we compute the histogram of the normalized stroke map, and then estimate the stroke width of text by searching for the peak value of the histogram. Fig. 2 gives the histogram statistics of NSW by our proposed algorithm.

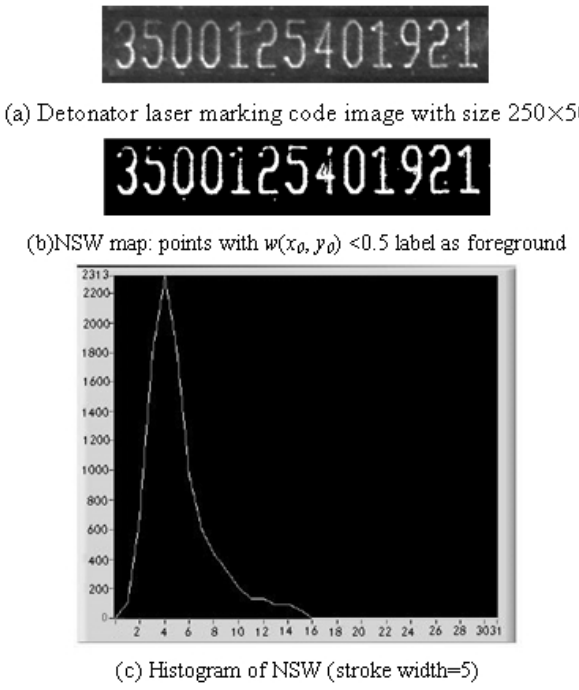

Fig. 2. NSW and histogram of stroke width in low quality images.

2.3 Text polarity determination

Among the foreground points, majority of points locate on stroke, only a small amount of them belong to backgrounds or stains, so we define the 0-1 NSW map as coarse binary image. To remove non-text parts and extract text stroke accurately, we firstly determine text polarity. Firstly, we compute the mean value of middle white pixels $\boldsymbol{WP}$ and outer contour pixels $\boldsymbol{CP}$ of NSW map in the gray scale image correspondingly. $\boldsymbol{WP}$ and $\boldsymbol{CP}$ are defined as follow:

$\boldsymbol{WP} = \{(x, y) \mid NSW(x, y) = 1, NSW(x \pm 1, y) = 1$ and $NSW(x, y \pm 1) = 1\}$

$\boldsymbol{CP} = (x, y) \mid NSW(x, y) = 0, NSW(x + (or -)1, y) = 0$ and $NSW(x + (or -)2, y) = 1$ or $NSW(x, y + (or -)1) = 0$ and $NSW(x, y + (or -)2) = 1$

After we obtain the number and position of $\boldsymbol{WP}$ and $\boldsymbol{CP}$, the mean value of the two pixel sets in the gray scale image are computed. The ratio is defined as:

$$R = \frac{m_{\boldsymbol{WP}}}{m_{\boldsymbol{CP}}} \tag{4}$$

Fig. 3. Background region removing.

The text polarity is bright when $\boldsymbol{R} > 1$ and dark when $\boldsymbol{R} < 1$. $\boldsymbol{R} = 1$ is an abnormal situation illustrating that the text region and background is inseparable. This specific situation has never happened in our practical experiment so we ignore it.

2.4 Text segmentation refinement

Since many text pixels in the intersection are missed and background regions may also be false extracted, the NSW map can only be regarded as coarse segmentation result. In this section, we address how to recall the missed text pixels and filter out the false extracted regions. Since we get the text polarity, a simple thresholding method can be applied to remove the background pixels. If a global threshold is chosen, it should be the normalized image mean m0. We set the white pixels, in which its value is less than m0, to 0. Shown in Fig. 3, in the renewed NSW map, the background region in the intersection of '4' is removed and the same to character '9'.

For recalling missed text parts, the searching work is performed superpixel by superpixel instead of pixel by pixel. We select simple linear iterative clustering (SLIC) superpixels [15] for text image segmentation. It's a local clustering of pixels in a constrained region and the distance measure enforces the compactness and regularity of its shape. We use gray level clustering instead of the Lab space online. Here we introduce the SLIC-superpixels in brief.

The simple linear iterative clustering (SLIC) performs in a given color space and x, y coordinates. This algorithm needs the input of desired number of superpixels firstly. Assume the number is $\boldsymbol{N}$, the image size is $\boldsymbol{w} \times \boldsymbol{h}$, the approximate size of each initial superpixel is $(\boldsymbol{w} \times \boldsymbol{h})/\boldsymbol{N}$ and the interval of neighbored superpixel center is $\boldsymbol{d} = \sqrt{(\boldsymbol{w} \times \boldsymbol{h})/\boldsymbol{N}}$. With the given number, size and interval, we can get the center of each superpixel in the image roughly. Then new center of each superpixel is computed as the average vector of all the pixels belonging to the cluster. The process is repeated iteratively until convergence. Instead of using simple Euclidean norm, we define a distance measure $\mathbf{D}$ as follows:

$$\begin{aligned} \boldsymbol{D}_i &= \|I - I_i\| \\ \boldsymbol{D}_{xy} &= \sqrt{(x - x_i)^2 + (y - y_i)^2} \\ \boldsymbol{D} &= \boldsymbol{D}_i + \frac{m}{d}\boldsymbol{D}_{xy} \end{aligned} \tag{5}$$

Where m is a compactness constant. The distance is computed within a $2\boldsymbol{d} \times 2\boldsymbol{d}$ region around each superpixel center on the xy plane. This measure, weighting the two distances: $\boldsymbol{D}_i$ and $\boldsymbol{D}_{xy}$, offers a good balance between gray value similarity and spatial proximity. We set the superpixel number $\boldsymbol{N} = (\boldsymbol{w} \times \boldsymbol{h})/(\boldsymbol{w}_s - 1)^2$ and this parameter will be discussed in the next section. $\boldsymbol{w}_s$ is the obtained stroke width. The result of performing SLIC to the example detonator image is shown in Fig. 4. Initially, we search the corresponding region on the renewed NSW map for each superpixel. If the space ratio of white pixels to black pixels are greater than 0.8, this superpixel is identified on the text stroke and defined as text superpixel. For other text superpixels' retrieval, we use a local region growing method [16] while the algorithm is performed on superpixel level instead of pixel level. For each non-identified superpixel, if one of its neighbors is the text superpixel, we compute the following intensity difference of two superpixels:

$$\triangle I = d_m + \frac{1}{2}d_a + \frac{1}{4}d_n \tag{6}$$

Definition 1 (Average intensity difference)

$$d_a = \left| \frac{\sum_{g_i \in r_i} g_i}{n_i} - \frac{\sum_{g_j \in r_j} g_j}{n_j} \right|$$

Where g_i, n_i denotes the intensity and number of pixels in superpixel r_i. This expression is used to compute the average intensity difference between two neighboring superpixels.

Definition 2 (Middle intensity difference)

$$d_m = |g_{mi} - g_{mj}|$$

Where g_{mi} is the middle value of intensity in superpixel r_i.

Definition 3 (Neighbored intensity difference)

$$d_n = \left| \frac{\sum_{g_i \in r_{\text{neighbori}}} g_i}{n_{\text{neighbori}}} - \frac{\sum_{g_{\text{neighborj}} \in r_{\text{neighborj}}} g_j}{n_j} \right|$$

Where $n_{neighbori}$, $n_{neighborj}$ are number of adjacent pixels in node r_i and r_j, $r_{neighbori}$, $r_{neighborj}$ are adjacent regions in two neighbored superpixels. d_n represents the average adjacent pixels intensity difference. If two neighboring superpixels both locate on the stroke, this value nearly equal to 0 for the continuity of character stroke. However, if one locates on stroke and the other one on background, the value of d_n is just as large as deviation in edge detection.

If the intensity difference $\triangle I < T$, the non-identified superpixel is labeled as text pixel. If $\triangle I \geq T$, the superpixel is labeled as checked superpixel. T is the intensity difference threshold which will be discussed in the next section. Then a repeat work is done on the non-checked superpixels until no new text superpixel is found. The final extraction result is shown in Fig. 4.

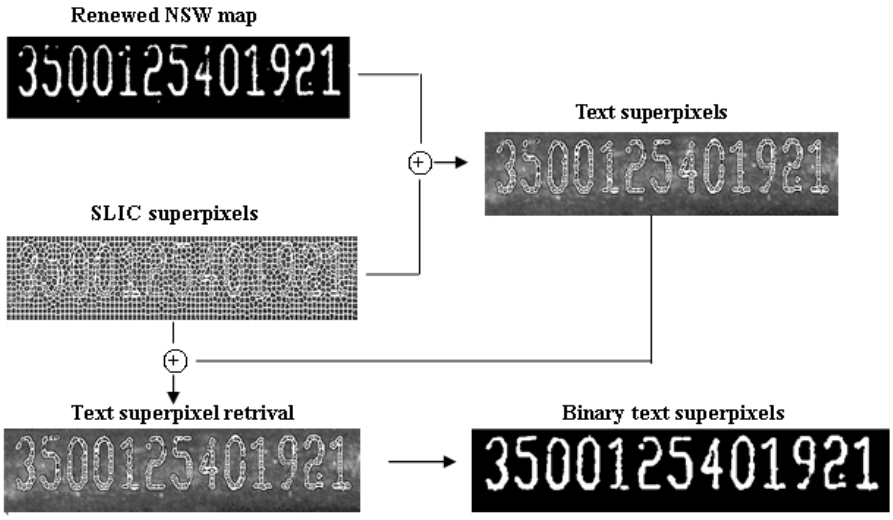

Fig. 4. Flowchart of text superpixel extraction.

3 Parameter analysis

Four important parameters in our framework are discussed in this section. All of them are selected adaptively demonstrating the robustness of our proposed method.

3.1 Intensity consistency threshold t

Considering that human visual response relates to local background luminance, the threshold t of intensity consistency comparison can be calculated as follows:

$$t = k \times mean(I)$$

Where k is constant coefficient and it is set to 0.8 in our experiment. *mean* () is average operator. It's a global threshold which will result in many text pixels missing due to the uneven illumination. Even if we use block strategy to select t locally, the result will be not improved much for a local growing method is implemented sequentially which will recall most lost strokes.

3.2 Surrounding window size r

This section discusses the relationship of text stroke width w and instantiated surrounding window size r. Example is shown below with r=20. The x axel is the position offset of text stroke and center of surrounding window. y axis is the normalized stroke width estimation.

From Fig. 5, we get that if $w < r/2$, the normalized stroke width will be less than 0.5 when the center of surrounding window is on the stroke. This feature

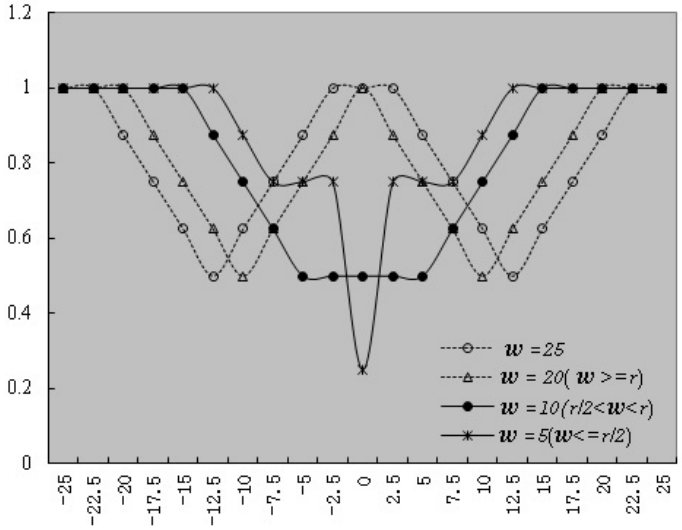

Fig. 5. Relationship of w and r.

can be used to extract text stroke. Therefore, we set $r > 2w$. w is estimated by simply applying Canny edge detection and finding the middle value of all parallel horizontal and vertical edges' width.

3.3 Initial superpixels number N

If $\boldsymbol{N}$ is smaller than a certain value, the segmentation may be blurred. Most of the superpixels located on the stroke contain too much background pixels. We call it coarse segmentation. The coarse segmentation cannot distinguish between background and character in superpixel level in the next step, for both of information may coexist in a single superpixel. If $\boldsymbol{N}$ is too much greater than the proper value, the segment result owns low under-segment error and high boundary recall [15].However, it is not profitable in time taken for the next step whose complexity is O($NlogN$). We consider it as over segmentation. If each superpixel in the image contains only one kind of pixels-background pixels or character pixels, it is fitting segmentation. In this situation, the stroke decomposed to less superpixels and each superpixel owns unambiguous information. In Fig. 6(b), three different sizes of superpixels are designed for the synthesized stroke and the red points represent the centers of seed superpixels around the stroke.We can find that the center may not fall in the stroke area if the interval d is greater than stroke width w_s in the left. In the middle is the situation $w_s/2 < d < w_s$, only one center locates on the stroke in the column array corresponding to horizontal stroke.If $w_s/2$ as shown in the right, more than one center of superpixels will fall in the stroke area. The three cases correspond to the three different segments in Fig. 6 (a). The fitting segment is the proper one and satisfies the defined condition. We choose number $\boldsymbol{N} = (\boldsymbol{w} \times \boldsymbol{h})/(w_s - 1)^2$.

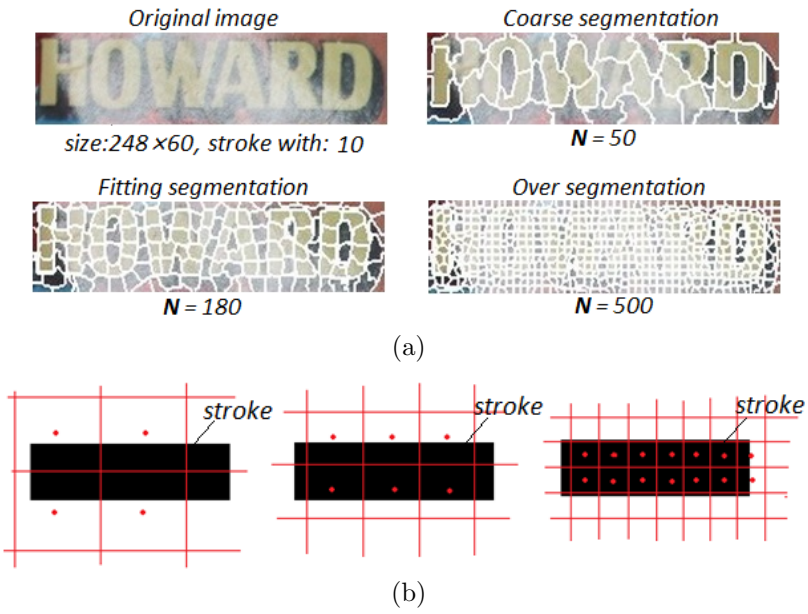

Fig. 6. Relationship between stroke width and superpixel numbers.

3.4 Intensity difference threshold T

As we know, the response of the eye to changes in the intensity of illumination is known to be nonlinear. The visual phenomenon depicted in Fig. 7 is the profile of the Just Noticeable Difference (JND) [17] in psychophysics concept.

JND is the minimum amount by which stimulus intensity must be changed in order to produce a noticeable variation in a sensory experience. Over a wide range of intensities, the expression is $\frac{\triangle I}{I} = \xi$.

$\triangle I/I$ is called Weber fraction, where the original intensity is I, $\triangle I$ is the addition amount required for the difference to be perceived (the JND), and ξ is a constant called the Weber constant. When out of this range, the JND is markedly changed with variant intensity of I. By experimentation, we chose $\xi = 0.05$ for all applications in this paper. Then the intensity difference threshold $T = \xi \times I$. I represents superpixel owning lower intensity value between two comparable superpixels.

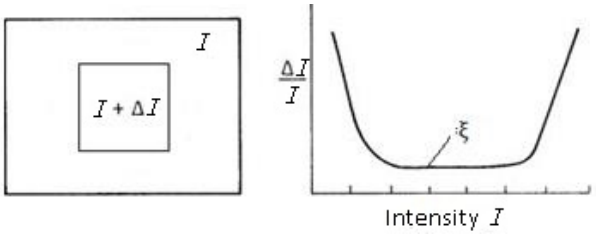

Fig. 7. Contrast sensitivity measurements.

4 Experimental result

In this section we compare the performance of our collected dataset with that of other classical text extraction methods published in the literature in their ability of binarization.

4.1 Data Collection

Since our method process images in low contrast with uniform background and text, the benchmark datasets are unsuited for our evaluation. Therefore, we select a collected dataset which consists of four different kinds of images. Those images are screenshot detected from detonator laser marking code, license plates, part of ICDAR 2011scene images and synthetic images as shown in Fig. 8.

Fig. 8. Four types images in our datasets.

The collected dataset has 4 types and totally 498 images including 212 detonator laser marking code images (***T*** 1), 164 license plates images (***T*** 2), 84 natural scene images (***T*** 3) and 28 synthetic images (***T*** 4). Those images suffer from different degradations, like uneven lighting, curved or/and shiny surfaces, low contrast to the backgrounds, different text fonts and sizes, different resolutions etc. As many as 4732 characters are contained in the test image. All candidate images are processed as black characters on white background for shown. Fig. 9 presents some examples of the binary results.

4.2 Result Comparison

We use two simple evaluation methods to verify the binarization performance [18] and OCR performance: the character extraction rate (***CER***) and the character recognition rate (***CRR***). They are defined as:

$$\boldsymbol{CER} = N_{segment}/N \qquad \boldsymbol{CRR} = N_{recognize}/N$$

Where $N_{segment}$ is the number of characters completely extracted without lost strokes or connecting to background, $N_{recognize}$ is the number of characters

correctly recognized by commercial OCR software ABBYY Fine Reader 9.0, and N is the total number of the characters.

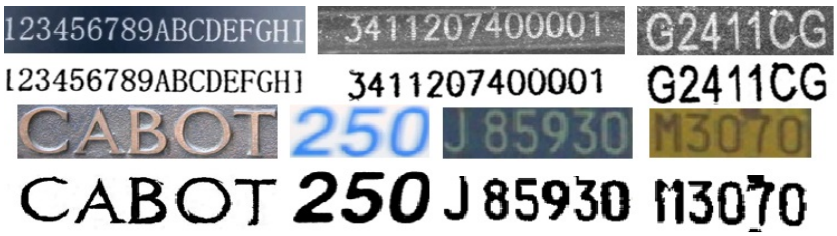

Fig. 9. Some examples of the segmented text.

Besides, we define some rules to judge whether the extracted character can be counted as correctly recognized characters for ***CER***: 1. An character which connects with large size background connected components is not counted. 2. An character with part missing should not be counted (missing size larger than 20% of the character).

To highlight the effectiveness of our proposition, we compare our proposed method with seven classical and competing binarization techniques:

– Otsu's global thresholding method derived from discriminated analysis [1].

– A local version of Otsu's method, where a sliding window is shifted across the image and the threshold is calculated from the histogram of the gray values of the window.

– Satish's edge based local method [5] which detects the edges in image and then passes a threshold surface through the edge pixels in a window locally.

– Niblack's method [2] which calculates a threshold surface $T = m + k \times s$ using the mean m and the variance s of the gray values by shifting a rectangular window across the image with parameter k = -0.2.

– X. Chen's local thresholding method [4]. It's a variant algorithm of Niblack's that selects window size adaptively.

–MSER based method [6]. It is widely used in text detection using the gray level stability of text region and background regions.

–MRF method [9]. It belongs to high level image processing method and the test includes training part.

Fig. 10 shows three images taken from the collected dataset and results of the five different binarization techniques and our proposed method. As can be seen in Fig. 10, Otsu's method suffers from the known disadvantages of global thresholding methods, *i.e.*. imprecise separation of text and background in case of blur or high variations of the gray values. The windowed version improves the segment precision, but creates additional non-text structures. As a result of the fact that the shift window may calculate the threshold even in areas where only background exists. The disadvantage of edge based method is that

it's very sensitive to noise and depends much on the quality of edge detection. Niblack's method has the same problem with local OTSU method, which segments the characters very well, but also suffers from noise in the zones without text. X. Chen's method, which is considered as the most successful binarization method recently, improves Niblack's method and performs better meaning while increases the computational time severely. MSER based method performs well for natural scene images, license plates images and synthetic images except detonator images. This arises from the uneven gray scale of text and low contrast in the detonator images. The MRF method is time consuming which takes 47 seconds on average to produce the final binary results but our method only takes 12 seconds on average with the same system. Besides, our proposed method combines text width and intensity similarity in superpixel level and can not only extracts text with the same stroke width but also can automatically filter out background spots that is smaller than text width. An evaluation of these binarization techniques using OCR are given in the Table 1.

As we can see, our method is comparable with the best binary algorithm that depicts in X. Chen's method. Although its binary performance is better from a

Fig. 10. Binarization results of different methods. (a) Global Otsu. (b) A local windowed version of Otsu. (c) Edge based local detection method. (d) Niblack's method. (e) X. Chen's local thresholding method. (f) MSER based method. (g) MRF based method. (h) Our proposed method.

human view, the OCR system does not give a better recognition result than our method for recognize text from low quality images.

Table 1. Performance for different binarization methods.

Bin. method	***CER***					***CRR***				
	(***T*** 1)	(***T*** 2)	(***T*** 3)	(***T*** 4)	Total	(***T*** 1)	(***T*** 2)	(***T*** 3)	(***T*** 4)	Total
Global OTSU	72.2%	92.6%	84.4%	76.5%	79.4%	69.5%	90.4%	63.5%	75.2%	74.9%
Local OTSU	81.6%	94.1%	79.5%	74.4%	83.8%	74.7%	90.2%	60.6%	70.2%	75.3%
Edge based	92.3%	93.5%	91.6%	92.9%	92.7%	90.3%	91.1%	86.4%	90.5%	90.2%
Niblack's method	85.2%	97.4%	85.0%	93.1%	89.0%	80.8%	96.3%	72.0%	90.5%	86.8%
X. Chen's method	94.8%	98.6%	92.7%	97.6%	96.4%	93.5%	96.9%	84.4%	96.4%	93.6%
MSER method	90.6%	92.0%	94.0%	98.3%	93.8%	86.5%	95.8%	87.2%	97.0%	91.4%
MRF method	97.4%	96.0%	94.0%	98.1%	96.8%	95.8%	94.5%	83.9%	97.8%	95.2%
Our method	97.3%	96.0%	92.2%	96.1%	96.2%	96.0%	95.2%	84.2%	96.0%	95.7%

5 Conclusion

In this paper, we present a new algorithm to segment text from low quality images. We consider that the text consists of wide lines and employ not only the intensity information but also the structure of the text stroke. Besides, a new text stroke width computation method is presented by using intensity similarity in a surrounding window. Utilizing the stroke width information, we propose a superpixel by superpixel growing method based on the text stoke width to recall missing text stokes. The presented algorithm is robust due to that all of the parameters are selected adaptively. In this paper, we collect our datasets including four types of low quality images suffering from different degradations. To demonstrate the effectiveness of our approach, a commercial OCR system is used to evaluate the precision performance and we achieve comparable result with those best binary algorithms.

The limitation of our method is that the it cannot be used in text images with various stroke width. And for complex background text segmentation, it proves to be no superior. Therefore, we will search for more general solution for text segmentation with more complex background in future work.

References

1. Otsu, N.: A threshold selection method from gray-level histograms. Automatica **11** (1975) 23–27
2. Trier, O.D., Jain, A.K.: Goal-directed evaluation of binarization methods. Pattern Analysis and Machine Intelligence, IEEE Transactions on **17** (1995) 1191–1201

3. Wolf, C., Jolion, J.M.: Extraction and recognition of artificial text in multimedia documents. Formal Pattern Analysis & Applications **6** (2004) 309–326
4. Chen, X., Yuille, A.L.: Detecting and reading text in natural scenes. In: Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on. Volume 2., IEEE (2004) II–366
5. Satish, M., Lajish, V., Kopparapu, S.K.: Edge assisted fast binarization scheme for improved vehicle license plate recognition. In: Communications (NCC), 2011 National Conference on, IEEE (2011) 1–5
6. Yin, X., Huang, K., Hao, H.: Robust text detection in natural scene images. (2013)
7. Mancas-Thillou, C., Gosselin, B.: Color text extraction with selective metric-based clustering. Computer Vision and Image Understanding **107** (2007) 97–107
8. Li, J., Tian, Y., Huang, T., Gao, W.: Multi-polarity text segmentation using graph theory. In: Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on, IEEE (2008) 3008–3011
9. Mishra, A., Alahari, K., Jawahar, C.: An mrf model for binarization of natural scene text. In: Document Analysis and Recognition (ICDAR), 2011 International Conference on, IEEE (2011) 11–16
10. Subramanian, K., Natarajan, P., Decerbo, M., Castañòn, D.: Character-stroke detection for text-localization and extraction. In: Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on. Volume 1., IEEE (2007) 33–37
11. Jung, C., Liu, Q., Kim, J.: A stroke filter and its application to text localization. Pattern Recognition Letters **30** (2009) 114–122
12. Epshtein, B., Ofek, E., Wexler, Y.: Detecting text in natural scenes with stroke width transform. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE (2010) 2963–2970
13. Fairchild, M.D.: Color appearance models. John Wiley & Sons (2013)
14. Liu, L., Zhang, D., You, J.: Detecting wide lines using isotropic nonlinear filtering. Image Processing, IEEE Transactions on **16** (2007) 1584–1595
15. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Susstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. Pattern Analysis and Machine Intelligence, IEEE Transactions on **34** (2012) 2274–2282
16. Liu, Q., Jung, C., Moon, Y.: Text segmentation based on stroke filter. In: Proceedings of the 14th Annual ACM international Conference on Multimedia, ACM (2006) 129–132
17. Shen, J.: On the foundations of vision modeling: I. webers law and weberized tv restoration. Physica D: Nonlinear Phenomena **175** (2003) 241–251
18. Shi, C., Xiao, B., Wang, C., Zhang, Y.: Adaptive graph cut based binarization of video text images. In: Document Analysis Systems (DAS), 2012 10th IAPR International Workshop on, IEEE (2012) 58–62