MULTI-POLARITY TEXT SEGMENTATION USING GRAPH THEORY

Jia Li [a], *Yonghong Tian* [b], *Tiejun Huang* [b], *Wen Gao* [a,b]

[a] Institute of Computing Technology, Chinese Academy of Science, Beijing, 100080, China
[b] Institute of Digital Media, School of EE & CS, Peking University

ABSTRACT

Text segmentation, or named text binarization, is usually an essential step for text information extraction from images and videos. However, most existing text segmentation methods have difficulties in extracting multi-polarity texts, where multi-polarity texts mean those texts with multiple colors or intensities in the same line. In this paper, we propose a novel algorithm for multi-polarity text segmentation based on graph theory. By representing a text image with an undirected weighted graph and partitioning it iteratively, multi-polarity text image can be effectively split into several single-polarity text images. As a result, these text images are then segmented by single-polarity text segmentation algorithms. Experiments on thousands of multi-polarity text images show that our algorithm can effectively segment multi-polarity texts.

Index Terms— Text processing, Image segmentation, Graph theory

1. INTRODUCTION

With the proliferation of images and videos, semantic analysis of these media is highly demanded for media content understanding and retrieval. Generally, text information such as caption and scene text is an important semantic clue for media content understanding. To extract text information from images and videos, text segmentation is an essential step. Given a text-line image which is usually generated from the text detection step, text segmentation, or named text binarization, can turn the literal content of this image into a collection of texts. Often, many texts in images and videos are multi-polarity, which means those texts with multiple colors or intensities in the same line.. It is thus necessary to develop effective and robust multi-polarity text segmentation methods.

Traditionally, there are three categories of text segmentation methods. In the first category, direct binarization methods utilizing global thresholds are adopted while predefining texts as light or dark. Among them, Niblack's method had been proved to give the best performance [4]. However, these methods would fail when the background is complex or text polarity is initially unknown.

Alternatively, some researchers try to determine text polarity with predefined rules. Lyu *et al.* [8] employed the color polarity classification methods proposed in [12] to determine the polarity of texts. In [7], texts were extracted based on stroke filters and their polarities were determined using a SVM classifier. In [14], a split-merge strategy was adopted to select connected components (CCs) corresponding to texts. Generally, this kind of methods can segment both light and dark texts but will fail when light and dark texts appear in the same text line.

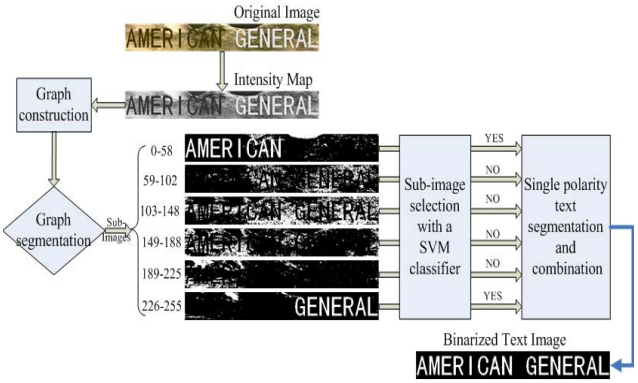

Figure 1.Flowchart of the text segmentation.

In the third category of text segmentation methods, temporal information were used when segmenting captions in videos. Chang *et al.* [1], Lienhart *et al.* [6] and Mi *et al.* [9] utilized temporal information to assist the segmentation of captions in videos. Although the segmentation performances were improved to a certain extent compared with text segmentation in images, their algorithms relied on the assumption that text polarity in a text line is fixed and won't change along with time. Therefore, when these three categories of methods are applied in multi-polarity text segmentation tasks, some important parts of multi-polarity texts will be discarded, thereby leading to incomplete results.

In this paper, we propose a novel algorithm to effectively segment multi-polarity texts. First, the intensity map of a color image is represented with an undirected weighted graph which is then partitioned into several sub-graphs representing continuous gray levels, thus corresponding to different color sub-images. In this step, multi-polarity text images will be split into several single-polarity text images so that texts with similar color will lie in the same sub-image. After that, sub-images containing texts will be selected out with a SVM classifier. Finally, by applying single polarity text segmentation methods on selected sub-images, texts with different polarities are extracted and then combined to form the final binarized text image. The flowchart of the algorithm is illustrated in figure 1.To get a clear view, foreground pixels in sub-images are shown in white and backgrounds are shown in black.

This paper is organized as follows: Section 2 presents a graph based method to divide multi-polarity text image into several sub-images with single-polarity texts. Section 3 discusses some simple post processing methods to get the final binarized text images. The experiments and results are presented in Section 4. Finally, the paper is concluded in section 5.

2. SUB-IMAGE GENERATION

To split a multi-polarity text image into several single-polarity text images, we have to find all possible color ranges corresponding to texts. To reduce the computational cost, we try to find the gray ranges instead. Given a color image, its intensity map can be represented with an undirected weighted graph by treating pixel groups at each gray level as nodes. Weights of edges linking these nodes are defined as correlations of these pixel groups. After that, sub-image generation can be formulated as a graph cut problem.

According to specific graph cut theory, this graph will be partitioned into several sub-graphs, each of which covers a specific gray range and denotes a sub-image possibly with single polarity texts. By using a SVM classifier, sub-images containing texts will be selected and the multi-polarity text segmentation problem is turned into several single-polarity text segmentation problems. More details are discussed as follows.

2.1. Graph Construction

In the graph, weights of edges are used to express correlations of pixel groups at different gray levels. Generally, correlation of pixel groups at gray levels C_i and C_j is affected by two factors:

1) the absolute difference of two gray levels $|C_i - C_j|$, and
2) the co-occurrence characteristics of these pixels. If a pixel at gray level C_i appears in the 8-connected neighborhood of a pixel at gray level C_j, there is a co-occurrence pair (C_i, C_j). $N_{i,j}$ is used to denote the number of such co-occurrence pairs.

Considering these two factors, correlation of pixel groups at two different gray levels can be expressed as in formula (1):

$$W_{i,j} = \begin{cases} 0 & if\ |C_i - C_j| > \mathrm{T_C}, i \neq j, \\ \exp\{-\dfrac{|C_i - C_j|}{\mathrm{T_C}}\} \times \sqrt{\dfrac{2N_{i,j}}{H_i + H_j}} & elsewise, \end{cases} \quad (1)$$

where H_i and H_j denote the numbers of pixels at gray levels C_i and C_j in the intensity map respectively. To reduce the computational cost, correlations of pixel groups with absolute difference of gray levels larger than $\mathrm{T_C}$ will not be taken into account. In this paper, we empirically set $\mathrm{T_C}$ to 16.

2.2. Graph Segmentation

To separate texts with different polarities, named texts at different gray levels, this graph should be partitioned into several sub-graphs representing different gray levels. In the partition, correlations between sub-graphs should be small and correlations inside each sub-graph should be great. There are already a lot of methods for graph partition, such as min-max cut [2], normalized cut [11] and ratio cut [13]. However, these methods are usually time-consuming. To speed up the graph partition, we assume that a single character will lie in consecutive gray levels. With this assumption, we only need to segment the graph into sub-graphs that have continuous gray levels.

In this paper, we adopt the criteria proposed in normalized cut [11] to evaluate the cost of partition. The cost of partitioning a graph G into two disjoint sets A and B ($A \cup B = G$) is expressed as:

$$Ncut(A,B) = \frac{\sum_{i\in A, j\in B} W_{i,j}}{\sum_{i\in A, j\in G} W_{i,j}} + \frac{\sum_{i\in A, j\in B} W_{i,j}}{\sum_{i\in B, j\in G} W_{i,j}}, \quad (2)$$

where $Ncut(A,B)$ is the partition cost. Based on this formula, the cost of partitioning graph G consisting of continuous gray levels C_i to C_j into sub-graphs A and B is redefined as follows:

($A=\{C_i,...,C_x\}$, $B=\{C_{x+1},...,C_j\}$, $C_x \in \{C_i,...,C_{j-1}\}$)

$$S(C_x) = \begin{cases} 0 & H_x > 0, \sum_{i\in A, j\in B} W_{i,j} = 0 \\ +\infty & H_x = 0 \\ Ncut(A,B) & elsewise \end{cases}, \quad (3)$$

After that, the min cost and its position $C_{\min}$ is found as:

$$S(C_{\min}) = \underset{C_x \in \{C_i, C_{i+1} \ldots C_{j-1}\}}{MIN} \{S(C_x)\} \quad (4)$$

To avoid over-segmentation, the partition at $C_{\min}$ would be carried out only if at lease one of the following rules is satisfied:

Rule 1: $S(C_{\min}) = 0$.

Rule 2: $|C_i - C_j| > \mathrm{T_C}$, and $\mathrm{STD} > 0.5 \times \mathrm{MEAN}$.

here, rule 1 means graph G consists of two initially irrelative sub-graphs. Rule 2 requires that graph G covers much enough gray levels and this partition is stable [11]. STD and MEAN can be calculated using formula (5) and (6) respectively:

$$MEAN = \sum_{C_x \in \{C_i \ldots C_{j-1}\},\ H_x \neq 0} S(C_x)/K_{i,j}, \quad (5)$$

$$STD = \sqrt{\sum_{C_x \in \{C_i \ldots C_{j-1}\},\ H_x \neq 0} (S(C_x) - MEAN)^2 / K_{i,j}}, \quad (6)$$

where $K_{i,j} = \left\| \left\{ C_x \middle| C_x \in \{C_i, \cdots, C_{j-1}\},\ H_x \neq 0 \right\} \right\|$ .With these rules, graph G is bipartitioned iteratively until every sub-graph is undividable. After that, color sub-images corresponding to pixels in each sub-graph are generated. Among them, sub-images containing texts will be selected.

2.3. Sub-Image Selection

The key issue in sub-image selection is to find sub-images containing texts. In this paper, a SVM classifier is adopted to distinguish sub-images with or without text. To get the training set and test set, we generate multi-polarity text images through the following procedure:

1) Collect 2500 Chinese, 1000 English and 500 Digit phrases. After that, print these phrases onto clear background to form the ground-truth. All phrases are divided into two sub-phrases at random position.
2) 100 background images containing no texts are selected from videos, web-images *etc.* After that, we lay the ground-truths on random positions of these backgrounds. The average intensity of the overlapping region is denoted as I_m. To get human discernable text images, RGB colors of the two sub-phrases are selected from $[0, I_m - \mathrm{T_C}]$ and $[I_m + \mathrm{T_C}, 255]$ respectively if $I_m \in [64,192]$, otherwise the two phrases will

have random RGB color in range $[I_m + T_C, 255]$ if I_m is less than 64 or in range $[0, I_m - T_C]$ if I_m is greater than 192 .

After that, 4000 multi-polarity text images are obtained and those containing indiscernible texts are regenerated until all text images are human discernable. Among them, 1000 Chinese, 400 English and 200 Digit text images are selected as the training set, while the others are used as the test set. All text images in training set are segmented with the proposed algorithm. By comparing the generated sub-images with ground-truths, we get 2448 positive and 3364 negative sub-images (sub-images covering less than 1% pixels of the training image are removed to get a balanced training). Here we adopt 8 features for SVM-based classification similar to those used in [14]:

1) Pixel distribution features. As shown in figure 2, percentages of pixels in consecutive subbands Z_1, Z_2…Z_6, denoted as f_1, f_2...f_6, are used to express pixel distribution features.

2) Variation of stroke widths. The scales (stroke widths) of all pixels in a sub-image are calculated using methods proposed in [14]. For each pixel, lines in horizontal, vertical and two diagonal directions are drawn, which are then cut into line segments by edges of the corresponding CC. The scale of a pixel is then defined as the shortest length of these four line segments. With the mean M_S and standard deviation STD_S of these scales, the variation of stroke widths in this sub-image is defined as STD_S / M_S .

3) Size of CCs. Usually texts are formed with large CCs. To express this feature, we use the percentage of pixels in CCs that are large than 16 pixels to evaluate whether there are many large CCs in the sub-image.

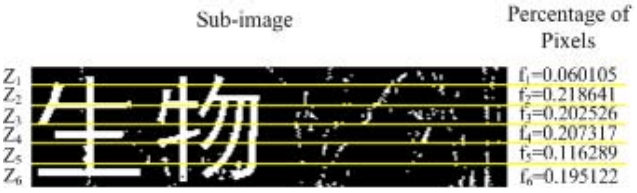

Figure 2.Illustration of pixel distribution features.

With these features and the SVM classifier, sub-images containing texts can be checked out. However, it is possible that none of the sub-images is selected. With the assumption that the input text image will surely contain texts, we simply select the sub-image with the largest (f_3+f_4).

3. POST PROCESSING

After the previous processing, several color sub-images containing single polarity texts are obtained. Then extraction of text from such sub-images can be treated as the traditional problem of single polarity text segmentation. Here we adopt some simple post-processing methods as follows:

1) Seed fill algorithm. As in [6], the pixels at the boundary of sub-images are treated as seeds to fill the pixels with similar colors, which will be further marked as backgrounds.

2) Hue histogram analysis. After graph segmentation and sub-image selection, backgrounds can be largely removed. Thus we can assume that most pixels in the sub-image are text pixels. By finding the greatest peak of the hue histogram and its corresponding two valleys, the hue range of text pixels is calculated. Pixels whose hues are not in this range are considered as backgrounds and will be removed.

3) Connected components analysis. Texts are usually formed by large CCs, thus CCs less than 16 pixels are removed. Also, CCs covering less than 10% area of its surrounding text box are removed too.

After these steps, each selected sub-image contains only text pixels with similar intensities and hues. These sub-images will be combined together to generate binarized text image.

4. EXPERIMENTS

Two test sets are used in our experiment. Test set I contains 2400 multi-polarity text images generated in subsection 2.3. Test set II contains 326 text images grabbed from web-images and videos.

To evaluate the effect of sub-image selection, we compare our algorithm with GMeans [3] and clustering algorithm used in [14] on test set I. K-Means algorithm with two predefined cluster numbers is also used for comparison. Text images from test set I are input into these algorithms and their generated sub-images are then classified using the methods proposed in subsection 2.3. If a selected sub-image covers more than 10% text area of the ground-truth, we say that this sub-image is correctly selected. Threshold here is set to be 10% since texts in images may have multiple colors, each of which covers a small percentage of the total text area. Accordingly, the number of correctly selected sub-images is defined as R_C, while the number of miss-selected and false-selected sub-images are defined as R_M and R_F. The recall and precision are defined as follows and results are shown in table 1.

$$\text{Recall}=R_C/(R_C+R_M), \quad (7)$$

$$\text{Precision}=R_C/(R_C+R_F), \quad (8)$$

Table 1. Effect of Sub-image Selection

	Recall (%)	Precision (%)
Our	81.11	**98.99**
GMeans[3]	78.50	79.86
Zhan[14]	62.37	97.59
K-Means(K=5)	77.82	80.44
K-Means(K=10)	**85.11**	54.87

In table 1, we see that our algorithm reaches the highest precision rate while K-Means algorithm with 10 clusters reaches the best recall rate. It is mainly because: (1) Colors in text image can be viewed as data clusters with significant overlap, while GMeans [3] usually overfit with much more clusters in this situation, thus leading to a low precision. (2) Performances of K-means algorithm with fixed K rely heavily on the predefined cluster number, which will not fit for all text images. It is obvious that the recall will increase and the precision will decrease along with an increasing K; (3) in our algorithm, spatial information is used to weight the similarity of pixels with different colors. With spatial information, the relation of different colors can be better

evaluated, which makes the segmentation more reliable. This leads to better recall and precision than the algorithm in [14] which only considered the similarity of colors.

Moreover, our algorithm can reach a higher speed than the other algorithms since text image has to be scanned only once. After that, pixels at the same gray level are treated as groups, which will greatly reduce the time cost in sub-image generation.

To demonstrate effects of our algorithm, some representative successful results are shown in figure 3 while some failures are illustrated in figure 4. All images are from test set I and II.

In figure 3, (a) and (b) contain single polarity texts with thin stroke widths and wide stroke widths respectively, which means our algorithm can adapt to different stroke widths. (c) and (d) are representative results of multi-polarity text segmentation for English and Chinese texts. It shows that our algorithm can well segment multi-polarity texts as well as single-polarity texts. In figure 4, some strokes in (a) are much thinner than other strokes and their colors are greatly affected by the background, which makes these strokes missed. In (b), background pixels have similar colors with texts, thus some texts are removed in the seed fill step in post-processing. In (c), there is only one letter with light polarity, which leads to a misclassification in the sub-image selection. In (d), texts with dark polarity have similar intensities with background pixels and they are failed to be separated in the graph segmentation step. Thus the sub-image containing this part of texts fails to pass the sub-image selection.

With respect to results above, we can find that main failures are induced by thin stroke widths and mistakes in sub-image selection, which will be solved in the future work.

Figure 3. Examples of successful segmentation.

Figure 4. Examples of failures.

5. CONCLUSION

In this paper, we propose a novel algorithm to segment multi-polarity texts. The main contribution of this paper is to propose an effective splitting strategy to segment multi-polarity texts into several single-polarity text images. An efficient sub-image selection strategy is also proposed which are proved to be useful in selecting sub-images containing texts. The effectiveness and robustness of our algorithm have been proved by experiments.

6. ACKNOWLEDGEMENT

This work is supported by grants from Chinese NSF under contract No. 60605020, National Hi-Tech R&D Program (863) of China under contract No. 2006AA01Z320 and 2006AA010105, and National Key Technology R&D Program under contract No. 2006BAH02A10.

7. REFERENCES

[1] Chang, F., Chen, G.C., Lin, C.C., and Lin, W.H. "Caption analysis and recognition for building video indexing systems". *ACM Multimedia Systems Journal, 10, 4*(Jan.2005), 344-355.

[2] Ding, C., He, X., Zha, H., Gu, M., and Simon, H. "A min-max cut algorithm for graph partitioning and data clustering." In *Proceedings of the IEEE International Conference on Data Mining*, 2001, 107–14.

[3] Greg Hamerly and Charles Elkan."Learning the k in k-means." In *Proceedings of the seventeenth annual conference on neural information processing systems* (NIPS), pages 281–288, 2003.

[4] Jain, A.K. "Goal-directed evaluation of binarization methods." *IEEE Trans. Pattern Anal. Machine Intell., 17, 12* (Dec. 1995), 1191-1202.

[5] Jung, K., Kim, K.I., Jain, A.K. "Text information extraction in images and video: a survey." *Pattern Recognition, 37, 5* (2004), 977-997.

[6] Lienhart, R. and Wernicke, A. "Localizing and segmenting text in images and videos." *IEEE Transactions on Circuits and Systems for Video Technology,12,4* (Apr 2002),256-268

[7] Liu, Q., Jung, C., and Moon, Y. "Text segmentation based on stroke filter." In *Proceedings of the 14th Annual ACM international Conference on Multimedia* (Santa Barbara, CA, USA, October 23 - 27, 2006). MULTIMEDIA '06. ACM Press, New York, NY, 129-132.

[8] Lyu, M.R., Song, J.Q., and Cai, M. "A comprehensive method for multilingual video text detection, localization, and extraction." *IEEE Trans on circuits and systems for video technology, 15, 2* (Feb. 2005), 243 – 255.

[9] Mi, C.J., Xu, Y., Lu, H., and Xue, X.Y. "A Novel Video Text Extraction Approach Based on Multiple Frames." *Fifth International Conference on Information, Communications and Signal Processing*, Dec. 2005, 678-682.

[10] Ostu, N. "A threshold selection method from gray-scale histogram." *IEEE Trans. Syst., Man, Cybern., vol 8* (Jan. 1978), 62–66.

[11] Shi, J., and Malik, J. "Normalized Cuts and Image Segmentation." *IEEE Trans. Pattern Anal. Machine Intell., 22, 8*(Aug. 2000), 888–905.

[12] Song, J.Q., Cai, M., and Lyu, M.R. "A robust statistic method for classifying color polarity of video text." In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (April 6-10, 2003). 581-584.

[13] Wang, S., and Siskind, J.M. "Image segmentation with ratio cut." *IEEE Trans. Pattern Anal. Machine Intell., 25, 6* (Jun. 2003), 675–690, .

[14] Zhan, Y., Wang, W., and Gao, W. "A Robust Split-and-Merge Text Segmentation Approach for Images." In *Proceedings of the 18th international Conference on Pattern Recognition - Volume 02* (August 20 - 24, 2006). ICPR. IEEE Computer Society, Washington, DC, 1002-1005