2011 International Conference on Document Analysis and Recognition

Scene Text Extraction by Superpixel CRFs Combining Multiple Character Features

Min Su Cho, Jae-Hyun Seok, Seonghun Lee and Jin Hyung Kim
Department of Computer Science, KAIST
291 Daehak-ro, Yuseong-gu
Daejeon, Korea
Email: {mscho, jhseok, leesh, jkim}@ai.kaist.ac.kr

***Abstract*—Features and relationships based on character color, edge, stroke and context plays a role for text extraction in natural scene images, but any single feature or relationship is not enough to do the job. This paper presents a novel approach for combining features and relationships within the Conditional Random Field (CRF) framework. By a simple homogeneity measure, an input image is over segmented into perceptually meaningful superpixels and then the text extraction task is formulated as a problem of superpixel labeling. Such a formulation allows us to achieve parameter learning from training images and probabilistic inferences by combining all the features and relationships of the input image. The proposed method shows high performance, in terms of quality, on both the KAIST scene text DB and the ICDAR 2003 DB.**

***Keywords*-scene text extraction; superpixels; character features; conditional random fields;**

I. Introduction

Analyzing the contents of camera-captured images has drawn a lot of attention with the wide spread of digital media. Among various contents in images, scene text recognition has been studied intensively since it can provide contextual information about the scene and can be easily exploited in further applications.

Three consecutive processes are usually performed to obtain the text information from an image. The first process is detection which localizes the text regions in the image. For each text region, extraction is then carried out to separate text components from the backgrounds in that region. Finally, the text information is obtained by OCR modules. Although each process depends on the performance of its previous process and therefore overall performance depends on the all three, the second step, scene text extraction, is the most essential process for accurate understanding of texts in camera-captured images. We focus on the second, the scene text extraction process, in this paper.

The intrinsic properties of text have been well exploited to extract text components from images [1–8]. Text components generally have a homogeneous color and they are separated from the background by a strong edge. They also have uniform stroke width and distinctive local shapes. However none of these features are robust enough to do the job by itself. Text colors and edge information are often corrupted by strong illumination, shadows and reflection. The stroke width tends to slightly vary even within the same character. Parts of characters are often confused with similar-looking background parts.

To overcome such subtleties in dealing with natural scenes, it is better to consider all the possible features and relationships obtainable. For this purpose, we chose to use the Conditional Random Field (CRF) modeling framework and formulated the text extraction task as a problem of pixel labeling into either text or background. CRF modeling allows us to use parameter learning from the training image database and also probabilistic inference by combining all the features and relationships in the input image.

Random field constructions with image pixels cause inevitably large number of nodes, and therefore, high computational complexity. Furthermore, observing meaningful edge information at the pixel level is difficult. To overcome these limitations, pixels in fairly homogeneous regions are grouped into superpixels and the CRF model is formulated by using the superpixels as nodes.

The rest of this paper is organized as follows: Section 2 provides a brief discussion on related works in scene text extraction. In section 3 we discuss what kinds of character features are useful and how to combine them in superpixel CRFs. The experimental results are given in Section 4, and conclusions are given in Section 5.

II. Related Work

Adaptive binarization methods have been widely used to extract text components. Gatos et al. [1] proposed to apply adaptive binarization on grayscale images and inverted grayscale images. To overcome the limitations of grayscale image handling, a number of techniques have been studied for the clustering and binarization in Lab, HCL and HSL color spaces [2–4]. However, the color variation in unrestricted environments make it difficult to accomplish the goal using a single color space. In response, Mancas-Thillou et al. [5] proposed selecting the most discriminative measure from two complementary distance measures.

Edge has also been exploited to extract text components [6, 7]. Ezaki et al. [7] presented the way of generating the connected components by binarizing both edge map and reversed edge map. However, this method often misses some targets because binarized edges of characters are not always closed.

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.209

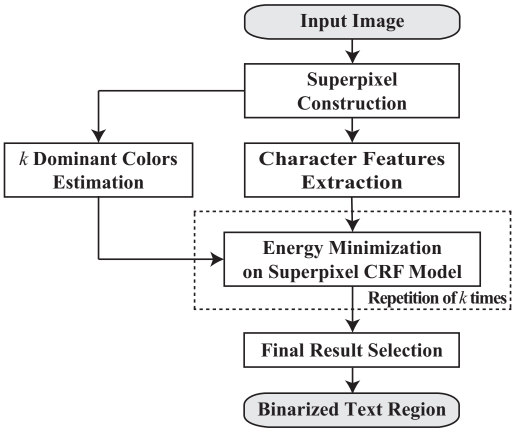
Input Image
Superpixel Construction
k Dominant Colors Estimation
Character Features Extraction
Energy Minimization on Superpixel CRF Model
Repetition of k times
Final Result Selection
Binarized Text Region

Figure 1. A flowchart of the proposed method.

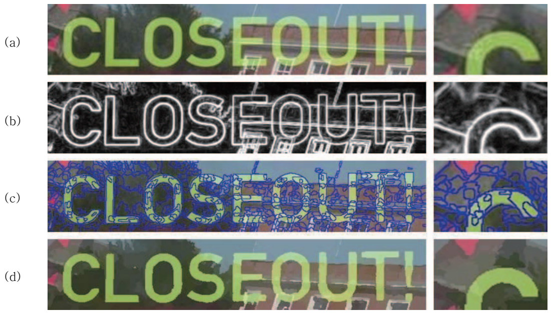

Figure 2. Procedure for constructing superpixels. (a) Input image. (b) Edge map of (a). (c) Superpixel image (boundaries of superpixels are marked as blue lines) (d) Quantized image with average colors of superpixels.

In contrast to the previous contributions, we propose a unified probabilistic framework which combines multiple character features to be useful for identifying text components.

III. Proposed Method

In this paper, we propose a scene text extraction system for isolating text components from natural scene images. We assume that text regions have previously been detected in the input image. As mentioned in the previous section, text extraction is a critical and essential step as it determines the quality of the final recognition result. This step also should consider uneven lighting and a complex background. We set the scene text extraction as a foreground/background labeling problem of the image superpixels.

Figure 1 shows the flowchart of the proposed method. First, the image is segmented into superpixels with a low-level grouping algorithm, with a simple measure of color. For each superpixel, computed are distinctive features that may be used to tell the character components from the background components. We call them character features, which could be color, edge strength, stroke width and contextual features.

Since a text is tightly situated within the detected text region, we may assume that the text color is one of the dominant colors. Dominant colors (i.e. text color candidates) are estimated by using k-means clustering. For each text color candidate, the label of a superpixel is inferred by the combination of the multiple character features and relationships among the superpixels. Among several label configurations, the final result is selected to have the minimal energy value.

A. *Superpixel Representation of Image*

A superpixel is a perceptually consistent unit such that all pixels in a superpixel are most likely uniform in color and texture. The number of superpixels is much smaller than the number of pixels in the image so the computational complexity of the problem can be reduced dramatically. In addition, more meaningful character features can be generated from superpixels. For instance, it is hard to observe robust edge features at the pixel level, but it is not difficult to do this at the superpixel level. Considering that text components are often separated from the background by strong edges, the set of superpixels are constructed by applying the watershed algorithm [9] on the color edge map. We can over-segment the image so that superpixels become as small as possible. By doing so, it can prevent a situation where a text part and a background part are merged into a superpixel.

After constructing the superpixels, the value of the features of a superpixel is set to the average of the member pixels. Figure 2 provides the procedure for constructing the superpixel image from the input image.

B. *Character Features*

A character feature is defined as a distinctive feature which may be used to distinguish character components from background components. We consider four character features including color, edge strength, stroke width and contextual features of character regions. Color, stroke width and contextual features are very consistent clues for extracting text components from natural scene images. Edge information also provides accurate boundaries. We will briefly explain how each feature plays a role in separating text components from complex background clutter.

A text line in an image is assumed to have similar colors dominant in that region. Therefore, color can be used to measure the probability that a given superpixel is text as well as the local affinity between adjacent superpixels. The system estimates dominant colors from the input image by using k-means clustering and treats each dominant color as the text color candidate. The number of clusters k is fixed to three, which is good to handle typically complex backgrounds. Since we assume that the colors of the inside of text region are almost same, one cluster is obviously a part of the text, another one is a part of the background, and the third one is either text boundary or background. When a superpixel has a similar color to the text color candidate, it has a high probability of belonging to a text component.

Edge can also be utilized as a criterion of separating texts from backgrounds. Generally, text components and backgrounds have strong edge magnitudes on their boundaries. An edge map is usually generated from a grayscale image. However, converting a complex natural scene into a grayscale image may weaken the edge strength. Therefore, we compute the color edge map from the RGB color image [10]. When two superpixels have strong edge magnitudes on their boundaries, we can determine that these two superpixels belong to different labels.

The text components have consistent stroke widths relative to backgrounds do. We use Gabor filter [11] to compute the stroke width. Gabor filter is a linear filter which produces the magnitude of the gradient for a selective orientation in a scalable spatial domain. The distance between parallel edges is measured from Garbor filter response, and the distance is assumed to be the stroke width. For each pixel, the minimum distance to the peaks in the filtered images for four different orientations (0°, 45°, 90°, and 135°) is computed as the stroke width. When two superpixels have similar stroke widths, these superpixels have a high probability of belonging to the same labels.

Finally, a contextual feature is computed to reflect the local shape of character components. Character components tend to have smoother and cleaner local shapes than backgrounds. The contextual feature of each pixel is measured by a two-step Adaboost algorithm which combines useful local features such as the Histogram of Gradients (HOG), Mean Difference Features (MDF) and the Standard Deviation (SD). Those texture features show excellent performance for classifying the text blocks and background blocks [12]. The first-step Adaboost classifier decides whether each pixel is text or not based on the HOG, MDF and SD features as in [8]. The second-step Adaboost learns the patterns of how text confidence is distributed in character regions by using the scores of the first-step Adaboost classifier. We call the scores of the second classifier as the contextual feature. Figure 3 shows an example of computing the contextual feature. Note that all features are extracted in the 7 x 7 window and the HOG, MDF and SD are extracted from both the original image and the smoothed image.

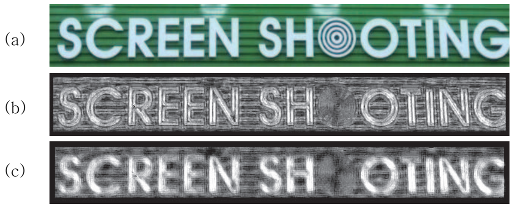

Figure 3. Example of contextual feature. (a) Input image. (b) Text confidence by local features. (c) Text confidence by contextual feature.

C. A Superpixel CRF Model

The proposed superpixel CRF model fuses the character features and learns the conditional distribution over the class labeling (i.e. text or background) given superpixels. A CRF model is an undirected graphical model which has the capability of unifying multiple features simultaneously in a single unified model. In the CRF model, the input image is represented as a 2-dimensional graph of superpixels, in which the feasibility of being text class for a single superpixel and the relationship of the neighboring superpixels are considered together.

In the CRF model, the conditional distribution of labels $\mathbf{x}$ given input features I is described as

$$P(\mathbf{x}|I,\Theta)=\frac{1}{Z(I,\Theta)}\exp(-E(\mathbf{x}|I,\Theta)),$$

where I represents the character features including the color, edge, stroke width and the contextual feature. $\mathbf{x}=\{x_i\}_{i\in S}$ represents binary labeling $x_i\in\{0,1\}$ which denotes background and text respectively, $Z(I,\Theta)$ is the partition function for normalization. The energy $E(\mathbf{x}|I,\Theta)$ of a configuration is linear in the model parameters $\Theta=\{\theta_{c_n},\theta_{sh_n},\theta_{c_p},\theta_{e_p},\theta_{sw_p},\theta_{sh_p}\}$.

The energy E is given by

$$E(\mathbf{x}|I,\Theta)=\sum_{i\in S}\psi_i(x_i,I,\Theta)+\sum_{i\in S,j\in ne(i)}\psi_{ij}(x_i,x_j,I,\Theta),$$

where S is the set of superpixels, and $ne(i)$ is the set of the adjacent superpixels of s_i.

The first term of the energy function is the node potential function for the superpixels, which represents the penalty of being the label for each superpixel. It is defined as

$$\psi_i(x_i,I,\Theta)=\theta_{c_n,x_i}f_{c_n}(x_i,I)+\theta_{sh_n,x_i}f_{sh_n}(x_i,I),$$

where the functions f_{c_n} and $f_{sh_n}(x_i,I)$ are given by

$$f_{c_n}(x_i,I)=\begin{cases} sigmoid(\frac{d(s_i,c_{text})-t}{\sigma}) & \text{if } x_i=1\\ sigmoid(-\frac{d(s_i,c_{text})-t}{\sigma}) & \text{if } x_i=0,\end{cases}$$

$$f_{sh_n}(x_i,I)=\begin{cases} 1-sh(s_i) & \text{if } x_i=1\\ sh(s_i) & \text{if } x_i=0.\end{cases}$$

$d(s_i,c_{text})$ is the Euclidean distance between the color of the superpixel s_i and the text color candidate c_{text}. If the distance between a color of a superpixel and the text color is close to zero, the superpixel may have a high probability of being text. However, it is difficult to apply the same criteria for all images, since each image has its own distribution of distances from the text color to the colors of the image pixels. Therefore, we intend to establish the appropriate criteria by Otsu's method [13] which reflects the statistic of the distances in each image. t is the threshold value and

σ is the mean of intra-class standard deviations, which are computed by Otsu's method.

$sh(s_i)$ is the likelihood that the superpixel s_i is a text component based on contextual features. The contextual feature function $f_{sh_n}(x_i, I)$ is designed to reflect the characteristics of character components by observing the distributions of edges and their score patterns. In short, features including the HOG, MDF and SD measure rough text confidence and some patterns of text confidence in the neighboring area can be found by contextual features.

The second term of the energy function is the pairwise potential function, which considers the local affinity between adjacent superpixels. The pairwise potential function represents the penalty score of a given set of two neighboring superpixels being one of three types (both text, both non-text, or one text and one non-text). It consists of four features: a color similarity, edge strength on a boundary, difference of stroke widths and difference of contextual features. The pairwise potential is defined as

$$\psi_{ij}(x_i, x_j, I, \theta) = \sum_{t \in T} \theta_{t, x_i, x_j} f_t(x_i, x_j, I),$$

where $T = \{c_p, e_p, sw_p, sh_p\}$ is a set of the feature types. The function f_t is given by

$$f_t(x_i, x_j, I) = \begin{cases} d_t(s_i, s_j) & \text{if } x_i = x_j \\ 1 - d_t(s_i, s_j) & \text{if } x_i \neq x_j, \end{cases}$$

where $d_{c_p}(s_i, s_j)$ denotes the degree of the separation between two adjacent superpixels based on the feature type. $d_{c_p}(s_i, s_j)$ is the Euclidean distance between the color of the superpixels in RGB color space, and $d_{e_p}(s_i, s_j)$ is the strength of the edge magnitude at the boundary between s_i and s_j. $d_{sw_p}(s_i, s_j)$ and $d_{sh_p}(s_i, s_j)$ are the difference of the stroke widths and the contextual features of s_i and s_j respectively. Note that the values of $d_t(s_i, s_j)$ are normalized to the range of 0 to 1. The reason for including the energy values between the different labels is that the contrast between the different labels is also an important feature for separating the text components from backgrounds. For example, if adjacent superpixels have large difference in their colors, they will have a high probability of having different labels.

Our proposed multiple cues are informative for scene text extraction. To integrate these cues, we applied the same approach as Ren et al. [14] and estimate the relative importance among them.

D. Inference and Parameter Estimation

Approximate inferences using loopy belief propagation [15] is used to obtain the most probable configuration for each image. Even if it is not guaranteed that loopy belief propagation produces a global optimum, it converges quickly and performs well in many cases.

To estimate the parameters of the model, we maximize the log-likelihood of the parameters over $\mathbf{X}$ taking each image of a training set as an i.i.d. sample. The training set is composed of the collection of D images and their ground truth labels which are denoted by $I = \{I_1, I_2, \cdots, I_D\}$ and $\mathbf{X}^* = \{\mathbf{x}_1^*, \mathbf{x}_2^*, \cdots, \mathbf{x}_D^*\}$ respectively. The set of random variables $\mathbf{x}_d$ corresponds to the d-th image I_d, and its ground truth labels is denoted by $\mathbf{x}_d^*$. The log-likelihood of the training set is

$$\log P(\mathbf{X}^* | \mathbf{I}, \Theta) = \sum_d \log P(\mathbf{x}_d^* | I_d, \Theta).$$

The maximum-likelihood estimates of the parameters are found by using the gradient descent method. Since our model is log-linear in the parameters Θ, we can easily obtain the partial derivative for each parameter. For example, the partial derivative with respect to the parameter θ_{c_n} for each image is

$$\begin{aligned} \frac{\partial \log P(\mathbf{x}^* | I, \Theta)}{\partial \theta_{c_n}} &= -\sum_{i \in S} \delta(x_i^*, 0) f_{c_n}(x_i^*, I) \\ &\quad + \sum_{i \in S} \sum_{x_i} \delta(x_i, 0) f_{c_n}(x_i, I) P(x_i | I, \Theta). \end{aligned}$$

We approximated the marginal probabilities by performing loopy belief propagation with the current parameters. The training process continues until the maximum difference of the gradients for the parameters is less than a threshold.

IV. Experiments

We evaluated our approach on 540 various images from the KAIST scene text database [4] and the ICDAR 2003 Robust Reading competition database [16]. These images consist of normal environment and special case images affected by strong illumination and complex backgrounds. The text regions in these images are manually cropped around their bounding box.

The evaluation was based on pixel-wise precision, recall and F-measure. Let T be the set of foreground pixels in the ground truth image and P be the set of foreground pixels in the predicted image. Precision p is $\frac{|P \cap T|}{|P|}$, recall r is $\frac{|P \cap T|}{|T|}$ and F-measure is $\frac{2 \times p \times r}{p + r}$. These measures are estimated for each test image, and the averages of them represent the performance of the method.

The performance of the proposed method was compared with that of Jung's method [4]. Jung's method is an adaptive binarization framework which binarizes the color-distance-map. For a fair comparison RGB color space is used instead of HCL color space to measure the distance between colors. We also evaluated a partial method of the proposed method to compare Jung's method and the CRF-based method in the same setting which uses the same features. The Color-CRF method is the partial method which only considers color in the energy function. The comparison of the Full-CRF method with the Color-CRF method can also measure the

Table I
THE PERFORMANCE FOR 540 SCENE TEXT IMAGES

	Precision	Recall	F-measure
Jung et al. [4]	0.870	0.813	0.841
Color-CRF	0.815	0.872	0.842
Full − CRF	0.841	0.883	**0.861**

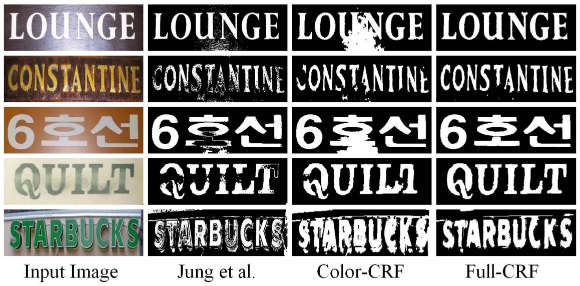

Figure 4. Examples of scene text extraction results.

effectiveness of additional character features (such as edges, stroke widths, and contextual features).

Table I compares the performance of these three methods. The Color-CRF method shows competitive performance with the Jung's method. This means that the CRF framework is as powerful as the adaptive binarization framework, while it allows considering multiple features at the same level. The superior performance of the Full-CRF method compared to the Color-CRF method shows that additional character features are effective and the the weights of the features are well adjusted during parameter estimation. Moreover, higher recall of the Full-CRF means that it can cover more cases than Jung's method with a small sacrifice of precision.

Figure 4 shows some examples of the scene text extraction result. It can be observed that many text components are extracted by combining color, edge and stroke widths. Moreover the extracted text components have shapes close to the real text shape in the image. Jung's method tends to discover the text components well, but it misses the parts of the exact text components in environments which have strong illumination and reflection. The Color-CRF method also produces inaccurate results when the background has a similar color to the text color because of the strong illumination and/or complex background. From these extraction results, we can see that combining multiple features of character components is important in order to extract the precise text components.

V. CONCLUSION

In this paper, a conditional random field model on a superpixel representation of images is presented to combine multiple features for scene text extraction. Our proposed method reduces the computational complexity by grouping pixels into superpixels. The weights of color, edge, stroke width and contextual feature of character components are adjusted according to the statistics obtained from training images, The combination of features provides the useful criterion to remove the ambiguity caused by complex backgrounds and strong illumination. We have evaluated the proposed method with various scene images, and observed that the segmentation results are more recognizable than when only color information is used.

ACKNOWLEDGMENT

This work was supported by the Industrial Strategic Technology Development Program (10035348, Development of a Cognitive Planning and Learning Model for Mobile Platforms) funded by the Ministry of Knowledge Economy(MKE, Korea)

REFERENCES

[1] B. Gatos, I. Pratikakis, K. Kepene, and S. Perantonis, “Text detection in indoor/outdoor scene images,” in *Proc. CBDAR*, 2005, pp. 127–132.

[2] A. Lai and G. Lee, “Binarization by Local K-means Clustering for Korean Text Extraction,” in *Proc. ISSPIT*, 2009, pp. 117–122.

[3] J. Yao, Y. Gao, L. Ma, and Y. Yang, “Scene Text Extraction Based on HSL,” in *Proc. ISCSCT*, vol. 2, 2008, pp. 315–319.

[4] J. Jung, S. Lee, M. Cho, and J. Kim, “Touch TT: Scene Text Extractor Using Touchscreen Interface,” *ETRI Journal*, vol. 33, no. 1, pp. 78–88, 2011.

[5] C. Mancas-Thillou and B. Gosselin, “Color text extraction with selective metric-based clustering,” *Computer Vision and Image Understanding*, vol. 107, no. 1-2, pp. 97–107, 2007.

[6] X. Liu and J. Samarabandu, “Multiscale edge-based text extraction from complex images,” in *Proc. ICME*, 2006, pp. 1721–1724.

[7] N. Ezaki, M. Bulacu, and L. Schomaker, “Text detection from natural scene images: towards a system for visually impaired persons,” in *Proc. ICPR*, vol. 2. IEEE, 2004, pp. 683–686.

[8] M. Li, M. Bai, C. Wang, B. Xiao, and Y. Lv, “Conditional random field for text segmentation from images with complex background,” *Pattern Recognition Letters*, vol. 31, pp. 2295–2308, October 2010.

[9] S. Beucher and C. Lantuéjoul, “Use of watersheds in contour detection,” in *Int'l Workshop Image Processing, Real-time Edge and Motion Detection/Estimation*, 1979, pp. 17–21.

[10] S. Di Zenzo, “A note on the gradient of a multi-image,” *Computer Vision, Graphics, and Image Processing*, vol. 33, no. 1, pp. 116–125, 1986.

[11] A. Jain and F. Farrokhnia, “Unsupervised texture segmentation using Gabor filters,” *Pattern recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.

[12] S. Hanif, L. Prevost, and P. Negri, “A cascade detector for text detection in natural scene images,” in *Proc. ICPR*. IEEE, 2008, pp. 1–4.

[13] N. Otsu, “A threshold selection method from gray-level histograms,” *Automatica*, vol. 11, pp. 285–296, 1975.

[14] X. Ren, C. Fowlkes, and J. Malik, “Cue integration for figure/ground labeling,” *Advances in Neural Information Processing Systems*, vol. 18, pp. 1121–1128, 2006.

[15] Y. Weiss, “Correctness of local probability propagation in graphical models with loops,” *Neural Computation*, vol. 12, no. 1, pp. 1–41, 2000.

[16] L. Sosa, S. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, “ICDAR 2003 Robust Reading Competitions,” in *Proc. ICDAR*, 2003.