Text Extraction from Scene Images through Color Image Segmentation and Statistical Distributions

Ranjit Ghoshal
St.Thomas'College of Engg.& Tech.
Kolkata-700023

Bibhas Chandra Dhara
Jadavpur University
Kolkata-700032

ABSTRACT

This article proposes a scheme for automatic extraction of text from scene images. We proceed by applying statistical features based color image segmentation procedure to the RGB color scene image. The segmentation separates out homogenous (in terms of color and brightness) connected components (CCs) from the image. We assume these CCs include text components. So, prime intention of this article is to inspect these CCs in order to identify possible text components. Here, a number of shape based features are defined that distinguishes between text and non-text components. Further, during learning, the distribution of these features are considered independently and approximate them using parametric distribution families. Here, we apply a selection for the best fitted distribution using likelihood criterion. The class (text or non-text) distribution is the multiplication of the corresponding feature distributions. Consequently, during testing, the CC belongs to the class that produces the highest class distribution score. Our experiments are on the database of ICDAR 2011 Born Digital Dataset. We have obtained satisfactory performance in distinguishing between text and non-text.

Keywords:

Scene Image, Color Image Segmentation, Connected Component, Statistical Distributions.

1. INTRODUCTION

Automatic recognition of text portions in a natural scene image is useful to blind and foreigners with language barrier. Such a recognition methodology should also employ an extraction of text portions from the scene images. Moreover, segmentation of such text portions have a crucial impact on document processing, content based image retrieval, robotics and intelligent transport systems. With the growing popularity of various image capturing devices such as digital cameras, mobile phones, PDAs etc, digital images are nowadays easily available. Extraction and recognition of texts from scene images captured by such devices is a challenging problem now-a-days. There have been several studies on text segmentation in the last few years. Wu et al.[9] use a local threshold method to segment texts from gray image blocks containing texts. By considering that texts in images and videos are always colorful, Tsai et al. [8] develop a threshold method using intensity and saturation features to segment texts in color document images. Lienhart et. al. [4] and Sobottka et. al. [7] use color clustering algorithm for text segmentation. In recent years, Jung et al. [3] employed a multilayer perceptron classifier to discriminate between text and non-text pixels. A sliding window scans the whole image and serves as the input to a neural network. High probability areas inside a probability map are considered as candidate text regions. Wavelet transform has also been applied for text segmentation. In this context Gllavata et al. [2] considered wavelet transform and K-means based texture analysis for text detection. Saoi et al. [6] improved the method of Gllavata et al. [2] and applied wavelet transform to all of R, G and B channels of input color image separately. More recently Bhattacharya et al. [1] proposed a scheme based on analysis of connected components (CCs) for extraction of Devanagari and Bangla texts from camera captured scene images. Also a few criteria for robust filtering of text components have been studied. In this article we first apply fuzzy c-means based clustering on the color image. With the assumption that text portions are homogeneous in color and lightness, different clusters may contain text portions as different connected components. The next step we follow is the study of these connected components. We define some features that are used to distinguish between text and non-text. We consider the text identification as a two class problem. Each class (i.e. text and non-text) is approximated by a combination of feature distribution. In the testing phase, we compare the score of each CC against these two classes. The CC belongs to the class with highest score. Concerning the database, we use the public database of ICDAR 2011 Born Digital Dataset.

2. COLOR IMAGE SEGMENTATION

Color image segmentation is our first step of text extraction. The fuzzy c-means algorithm is used for color image segmentation. Before applying fuzzy c-means we extract some features from the normalized RGB image. Let us consider a pixel p_i of the image. Then p_i can be described by the tuple (r_i, g_i, b_i) i.e. the normalized R, G and B values. Besides, these three color values, we take another two statistical features. Each of the statistical features considered here responds differently to different properties of text. Now, in the following we describe each statistical feature.

Statistical Feature1(s) : Let H is a greylevel histogram over a 7×7 window. The variance of H at each pixel is used to measure local information. It is defined as:

$$s = \sum_{j=1}^{N} (H(j) - \bar{H})^2. \quad (1)$$

Where $\bar{H}$ is the mean intensity of histogram H, and $N = 49$ is the number of pixels in the window.
Statistical Feature2(t) : Generally text regions have a high density of edges. This density is measured in a 13×13 window by summing all edge magnitudes located with a Sobel filter:

$$t = \sum_{j=1}^{M} E(j). \quad (2)$$

where $E(j)$ is the edge magnitude at pixel j, and $M = 169$ is the number of pixels in the window. Combining, the feature vector ($\mathbf{f}_i$) corresponding to the pixel p_i is: $\mathbf{f}_i = (r_i,\ g_i,\ b_i,\ s_i,\ t_i)$. These features are sent to the fuzzy c-means clustering procedure.

3. CONNECTED COMPONENT ANALYSIS AND FEATURE EXTRACTION

The image segmentation produces a number of connected components that are spread over several clusters. These components include the possible text portions. So, we now analyze these components to identify text portions. We assume, a single text component is homogeneous in terms of color and lightness. This assumption ensures that a single text component is not broken after clustering. Now, after segmentation normally the text parts make one single cluster. However a single cluster may have both text and non-text parts. This is when some text and some non-text region in the image have the same color. The connected components are now obtained from each cluster. Sufficiently small and large components do not contribute much for text identification. Small components generally represent noise and the large components are background. So such components are first removed. Further, we extract the following connected component (CC) based features to distinguish between text and non-text portions.

AR: The aspect ratio $AR = (height/width)$ of a non-text component is either very small or very large compare to text components.

OBR: The Object to background pixels ratio (OBR) is computed by taking the bounding box. Due to the elongated nature of texts only a few object pixels fall inside text bounding box. On the other hand, elongated non-texts are usually straight lines, so, contribute enough object pixels.

ER: The text like patterns are usually elongated. For the elongatedness ratio (ER) we use the measure designed by Roy et. al. [5].

TH: Thickness (TH) of a CC is calculated as: let h_i and v_i be the horizontal and the vertical run lengths of a pixel p_i at the i^{th} position of a component CC_j. We next compute the minimum of h_i and v_i and further constitute a set $MIN_j = \{m_i \text{ s.t. } m_i = \min(h_i, v_i),\ \forall i\}$. Thus MIN_j denotes the set of all the minimum run lengths considering all the pixels of CC_j. The thickness TH_j of the component CC_j is defined to be the element whose frequency is maximum of the set MIN_j.

MIH: Minimum size of the holes present in the CC. Generally, very small holes are presented in non-text components.

MXH: Maximum size of the holes present in the CC. Usually, text components have big holes compare to their sizes.

Combining these features, we construct the feature vector $\vec{\Upsilon} = \{AR, OBR, ER, TH, MIH, MXH\}$ for a connected component.

4. TEXT IDENTIFICATION METHODOLOGY

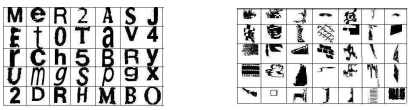
After obtaining all the features, we now intend to design a statistical model that takes into account the distribution of the features. The most popular such a model is the gaussian mixture model. However, the feature distributions may not always follow the Gaussian and hence a Gaussian mixture model may be ill-fitted. Non-Gaussian mixture distributions, on the other hand, are not yet very explored. Instead, we consider each feature independently. In other words, we apply a parametric distribution to each feature independently. Hence, we do not need mixture models. Here, we use an archive of distributions consisting of the Beta, Gaussian, Gamma, Log-Normal and generalized extreme value (abbreviated by gextreme) distributions. For each individual features, we apply each of these distribution. The best fitted distribution maximizes the likelihood value. This way we may choose the most appropriate distribution that describes a feature.
Each of the six different features has its own parametric distribution obtained by the above policy. Now we have two different classes, i.e. text class (C_{text}) and non-text class ($C_{non-text}$). For each of these classes, we could obtain six feature distributions. Combining all the six distributions for a class (say for C_{text}), we define the class distribution as:

$$f(C_{text}) = \prod_{i=1}^{6} f_i. \quad (3)$$

Here, f_i is the distribution corresponding to the i^{th} feature. Note, all $f_i, i = 1, \ldots, 6$, may not belong to the same family. Finally, we could assign a component to a class for which Eq. 3 gives the maximum value.

5. RESULTS AND DISCUSSION

Let us now present the segmentation and text extraction results. As for the database is concerned, we use the public database of ICDAR 2011 Born Digital Dataset. For the experiments, we select a total of 200 scene images, randomly from this database. Further, apply the fuzzy c-means algorithm for segmenting the images into a set of homogenous connected components. A few images and the corresponding segmentation results are presented in Table 2. We observe, our segmentation could preserve the text like components. Now, in order to construct the training data set, we manually label the connected components of the training sample images as text or non-text. Some of the images for the two classes (C_{text} and $C_{non-text}$) are shown in Fig. 1. The training set we construct have



(a) (b)

Fig. 1. (a) Text components. (b) non-text components.

4000 samples of text and 10000 samples of non-text components. We first perform a five-fold cross validation over the training data to assess the performance of our model. We get the cross validation accuracy 61.63% for C_{text} and 69.51% for $C_{non-text}$. Later, we obtain the distributions for the training and test classes. The individual features and their corresponding distributions are presented in Table 1. Finally, after obtaining the distribution we now provide

Table 1. Parametric distribution correspond to individual features.

Feature	C_{text}	$C_{non-text}$
AR	gextreme $(\xi = 0.1677, \sigma = 0.3852, \mu = 1.1072)$	gextreme $(\xi = 0.4717, \sigma = 0.5276, \mu = 0.4528)$
OBR	gextreme $(\xi = 0.3531, \sigma = 0.4172, \mu = 0.8683)$	gextreme $(\xi = 0.5143, \sigma = 0.7641, \mu = 0.9342)$
ER	Log-Normal $(\alpha = 2.1675, \beta = 0.0587)$	gextreme $(\xi = 0.2307, \sigma = 2.4130, \mu = 7.6359)$
TH	Log-Normal $(\alpha = 3.5317, \beta = 0.8148)$	gextreme $(\xi = 0.5231, \sigma = 7.5163, \mu = 10.8324)$
MIH	gextreme $(\xi = 5.0003, \sigma = 0.0037, \mu = 0.0021)$	gextreme $(\xi = 2.7377, \sigma = 2.5060 \times 10^{-7}, \mu = 7.9171 \times 10^{-8})$
MXH	gextreme $(\xi = 4.7340, \sigma = 0.2351, \mu = 0.0579)$	gextreme $(\xi = 2.6532, \sigma = 2.7455 \times 10^{-7}, \mu = 9.3706 \times 10^{-8})$

Table 2. Some images (first column), the corresponding segmentation results (second column) and extracted text components (third column). Often, some non-text components are included in text class and we missed some text components in some of the images.

the test samples. We put each connected components against the two classes and assign a component in the class C for which $f(C)$ (Eq. 3) is the maximum. Some of the images are the extracted text components (third column) are shown in Table 2.

6. SUMMERY AND FUTURE SCOPE

This article provides an automatic extraction of visual text entities embedded in scene images. It is based on color image segmentation followed by an extraction of several connected component based features that lead towards identification of text components. The features are considered individually, and are approximated by chosen parametric distributions. The class distribution score becomes the multiplication of the feature distributions. A component is assigned to the having the maximum score. We obtain satisfactory experimental results. One primary task of this article is to propose a way to describe feature distribution using statistical models. In this regards, this work may be considered as a starting point. We observe different families of distribution could best approximate the features. The correlation among the features is omitted here. However, linear or non-linear correlation may be present among the features. Thus the future task may be to propose a suitable model that incorporates correlation among the features or in other words among different distribution families.

7. REFERENCES

[1] U. Bhattacharya, S. K. Parui, and S. Mondal. Devanagari and bangla text extraction from natural scene images. In *Proc. of the Int. Conf. on Document Analysis and Recognition*, pages 171–175, 2009.

[2] J. Gllavata, R. Ewerth, and B. Freisleben. Text detection in images based on unsupervised classification of high frequency wavelet coefficients. In *Proc. of Int. Conf. on Pattern Recognition*, volume 1, pages 425–428, 2004.

[3] K. Jung, I. K. Kim, T. Kurata, M. Kourogi, and H. J. Han. Text scanner with text detection technology on image sequences. In *Proc. of Int. Conf. on Pattern Recognition*, volume 3, pages 473–476, 2002.

[4] R. Lienhart and F. Stuber. Automatic text recognition in digital videos. In *Image and Video Processing IV, Proc. SPIE 2666*, pages 180–188, 1996.

[5] A. Roy, S. K. Parui, A. Paul, and U. Roy. A color based image segmentation and its application to text segmentation. In *Proc. of Ind. Conf. on Computer Vision, Graphics & Image Processing.*, pages 313–319, 2008.

[6] T. Saoi, H. Goto, and H. Kobayashi. Text detection in color scene images based on unsupervised clustering of multihannel wavelet features. In *Proc. of Int. Conf. on Doc. Anal. and Recog.*, pages 690–694, 2005.

[7] K. Sobottka, H. Bunke, and H. Kronenberg. Identification of text on colored book and journal covers. In *Proc. of the Int. Conf. on Document Analysis and Recognition*, pages 57–63, 1999.

[8] C.M. Tsai and H.J. Lee. Binarization of color document images via luminance and saturation color features. *IEEE Trans. on Image Processing*, 11(4):434–451, 2002.

[9] V. Wu, R. Manmatha, and E. M. Riseman. Textfinder: An automatic system to detect and recognize text in images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(11):1224–1229, 1999.