Neurocomputing 175 (2016) 652–661

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Brief Papers

Scene text localization using edge analysis and feature pool

Chong Yu, Yonghong Song, Yuanlin Zhang

Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an 710049, China

ARTICLE INFO

Article history:
Received 2 July 2015
Received in revised form
30 August 2015
Accepted 28 October 2015
Communicated by Gaofeng Meng
Available online 9 November 2015

Keywords:
Text detection
Feature pool
Edge analysis

ABSTRACT

Due to the rapid development of machine learning and data mining in nowadays, how to acquire information from images becomes more and more important. The direct information of an image is the texts inside. However, detecting such texts in images is always a challenging problem in computer vision area.

Edge is one of the most important clues in scene character detection task. However, many edge based text detection methods usually had trouble with sticky edges and did not fully utilize characteristic of texts. In this paper, we proposed a method for detecting and localizing texts in natural scene images, by edge recombining, edge filtering and multi-channel processing. In order to segment texts from backgrounds, edges are firstly over-segmented into edge segments during edge analysis. These edge segments are then recombined to candidate characters and an edge filter is used to filter out most of background edges. The left candidate character edges are linked up to candidate text lines. We use two different classifiers to filter out non-text lines. To classify more accurately, extracted edge-based and region-based features are firstly stored in feature pools. Then we use liner SVM to select the most effective features from the feature pool to train classifiers. Finally, multi-channel is used to ensure the recall and a modified non-maximal suppress is applied to eliminate duplicate results. Experimental results on the ICDAR 2011 competition dataset and SVT database demonstrate the effectiveness of our method.

1. Introduction

Detecting and localizing text in scene pictures is always a challenging problem in computer vision area. Different from text in scanned materials, it is more difficult to locate the texts in natural scenes since there are much more complicated backgrounds and the texts are usually in multi forms [1]. Text localization is especially important in scene images since only with the accuracy location of texts, Optical Character Recognition (OCR) could recognize them efficiently.

Text in a local area generally have a homogeneous color, uniform stroke width and distinctive local shapes. Besides, most of text can be separated from the background by a strong edge. These properties have been exploited in many existing methods and proved effective. However, these features are not so useful all the time because of the influence of illumination, shadows and reflection. Apart from these influences, some background patterns like windows and bricks are so similar to text that many features are useless in distinguishing them.

In this paper, we describe an edge based text localization method with all these intrinsic properties fully utilized. Inspired by the progress in recent object detection and recognition research, our method intends to extract and label edges for each character. In addition, we apply several classifiers to filter out false detections. Different from many text detection methods, in our method efficient features are selected from a feature pool for training and predicting according to the decision function of liner SVM. Finally, to overcome the dependence on edge extraction, we use a multi-channel processing to extract edges in different conditions and fuse the results by a modified non-maximal suppress. The flowchart of the proposed method is given in Fig. 1. Under the standard ICDAR 2011 dataset and protocol [2], the proposed method achieves state-of-the-art results in text localization.

The rest of this paper is organized as follows. In Section 2 a brief discussion on related works is provided. Section 3–6 describe the proposed method. The experimental results are given in Section 7, and conclusions are given in Section 8.

2. Related work

Text detection in natural scene has been widely studied in recent years. The existing methods can be generally categorized into two groups – methods based on connected component analysis and methods based on region analysis. Methods in the first group usually obtain candidate characters by segmenting image into connected components. Then the candidate characters are classified directly or linked up to candidate text lines to analyze. One crucial problem of these methods is that text must be segmented from background before analysis. Methods of the latter

http://dx.doi.org/10.1016/j.neucom.2015.10.105
0925-2312/

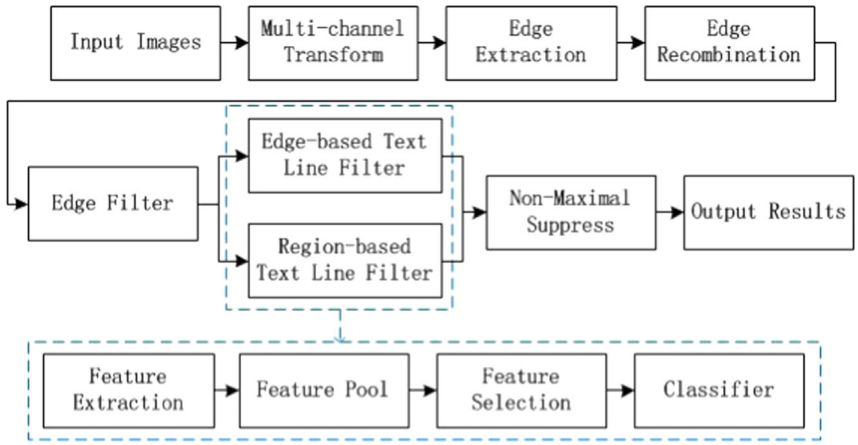

Fig. 1. Flowchart of the proposed method.

group usually depend on a slide window to detect a certain region and classify based on the region. Due to the large quantity of windows, these methods often take more time and quite depend on the accuracy of classifier.

Majority of recent published methods fall into the first group. These methods may also be subdivided into several classes according to the connected component unit. Local binarization algorithm is used to segment candidate CCs from gray-level image in Pan's method [3]. Neumann [4] and Chen [5] use MSER [6] to segment text from its around backgrounds. Since Epshtein proposed a text detection method using stroke width transform [7], many methods based on edges are published. Among them, Huang [8] filter out complex backgrounds by utilizing the essential spatial layout of edges and Meng [9] uses an energy function to calculate stroke width more precisely.

Region based methods usually use bottom-up approaches to generate candidate detection regions. Shivakumara [10] proposed a method using Laplacian filter in frequent domain to generate text regions. In another method, Wang and Babenko [11] use sliding windows to detect candidate regions in different image sizes and use Random Ferns to recognize characters directly from candidate regions. Similarly, Wang et al. [12] applied Convolutional Neural Networks to text classification after candidate regions generated by sliding windows.

In text classification, man-made features like HOG [13] are usually used, however, they are not always so efficient in some situations. To improve the accuracy of classification, many methods are proposed concentrating on feature construction. In [14,15], features generated by different block partitions are used and both perform well in text classification. Instead of regular features used in published methods above, Lee [16] applies liner SVM to select features in a feature pool in character recognition, which results out better than normal features. Due to the development of deep learning algorithm, a few text detection methods introduce Convolutional Neural Networks (CNNs) [12] in text classification and acquires good results.

3. Edge recombination

Different from many edge based methods, the extracted edges are not used directly in our method. One major reason is that in the images with complicated background, edges of texts are usually linked with some background edges. Using edges in these situations to analyze is very coarse and inaccuracy and text detection task easily fails. In our method, the extracted edges are firstly over-segmented into smaller edge segments using detected corners. Then the edge segments are recombined together by rules which make sure edges of the same character become one edge unit. In this way, text edges can be separated from background edges even there are complicated backgrounds.

3.1. Edge over-segmentation

Edge map is firstly extracted using Canny operator. The threshold of Canny operator is set experimentally so that most text edges can be reserved. One challenge of edge based methods is the segmentation of a different objects' edge. If the boundaries of texts and backgrounds cannot be segmented, the further classification could fail. In our method, to label the boundary of texts more precisely, edge map is over-segmented using Harris corners and intersections between edges (Fig. 2). The intersections are those pixels with two neighbor pixels of different directions. The direction of each pixel is calculated using gradient map.

3.2. Edge merging

After edge over-segmentation, boundaries of texts and backgrounds are segmented as different components. But meanwhile, the boundary of one character is also segmented into multi edge components. Therefore, the similar edge segments are then recombined as edge units using a neighbor map [9]. Briefly, the distance of two neighbor edge segments in many aspects such as color is calculated and merged if the distance is larger than the threshold. After this step, most edge segments that belong to the same character can be merged successfully. However, there are still some special situations, for example, character 'A' could be merged into two different units: the triangle edge inside and outside boundary. So we search from these edge points along gradient direction. If the edge is a part of character, we can find corresponding edge points and they share similar search length. The qualified edges are then merged again. An example of edge merging is shown in Fig. 3. From the result, we can see that most edges of different characters are labeled as separate units and the boundary of the same character is merged as a whole unit.

4. Edge filter

After edge recombination, edges of a character are labeled as a whole segment, so we can extract features based on the edge map. Considering characters in a text line usually share the same color and size, we use the features below to train a classifier and filter part of background edges. Note that properties such as width of