Neurocomputing 168 (2015) 970–978

Contents lists available at ScienceDirect

Neurocomputing

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

Text string detection for loosely constructed characters with arbitrary orientations

Yong Zhang [a,c], Jianhuang Lai [a,c,*], Pong C. Yuen [b]

[a] School of Information Science and Technology, Sun Yat-Sen University, China
[b] Department of Computer Science, Hong Kong Baptist University, Hong Kong, China
[c] Guangdong Key Laboratory of Information Security Technology, Guangzhou, China

ARTICLE INFO

Article history:
Received 14 April 2014
Received in revised form
24 January 2015
Accepted 8 May 2015
Available online 16 May 2015

Keywords:
Text detection
Text string
Stroke width transform
k-Nearest neighbors
Complex character

ABSTRACT

Text in a scene provides vital information of its contents. With the increasing popularity of vision systems, detecting general text in images becomes a critical yet challenging task. Most existing methods have focused on extracting neatly arranged text string for compactly constructed characters. Motivated by the need to consider the widely varying forms of scene text, we propose a stroke-based text detection method which detects arbitrary orientations text strings with loosely constructed characters in images. Our approach employs result of stroke width transform (SWT) as basic stroke candidates. These candidates are then merged using adaptive structuring elements to generate compactly constructed characters. Individual characters are chained using *k*-nearest neighbors algorithm to identify arbitrary orientations text strings, which are subsequently separated into words if necessary. To better evaluate our system and compare it with other competing algorithms, we generate a new dataset, which includes various characters and text strings in diverse scenes. Experiments on ICDAR datasets and the proposed dataset demonstrate that our approach compares favorably with the state-of-the-art algorithms when handling arbitrary orientations text strings and achieves significantly enhanced performance on loosely constructed characters in scenes.

1. Introduction

With the rapid development of digital technology and the broad use of Internet, the amount of multimedia resources such as image and video is increasing dramatically. Collating these multimedia data based on their contents is a big challenge for people. Manual labeling process is laborious and time consuming. It is desirable to develop the automatic methods for annotation and indexing of these multimedia resources.

A variety of techniques have been put forward to extract the content information from image [1]. Among these techniques, text detection is an effective and efficient method because of the rich information contained in the caption and scene [2]. Although many text detection approaches have been proposed in the last decades [3], detecting general text in scene is still a challenging task because the algorithms often suffer from multilingual text [4], orientation [5], and complexly arranged text string [6].

A text detection process is usually divided into three steps, namely, stroke detection, character extraction, and text string aggregation. Because different languages may have quite different appearances, few methods can handle multilingual text well. Some stroke-based methods claim the multilingual ability in the stroke detection step [7]. However, the stroke differences of different languages are not as critical, whereas the structural differences of them are very critical since they strongly affect the results of character extraction and text string aggregation. From the linguistic point of view, alphabetic literal such as English, Spanish, and German usually has compactly constructed characters, whereas ideograph such as Chinese, Japanese, and Korean is generally constituted by loosely constructed characters. The connected component (CC) based text detection methods [8–10] provides us with ample evidence to support this point. Alphabetic literal characters do not contain many connected components (CCs), this will not affect the character/letter extraction too much. However, ideograph characters contain many CCs and these structures increase the difficulty of character extraction. This fact can be demonstrated by Fig. 1(a), in which we use English and Chinese as an example.

Another major challenge of existing stroke-based text detection methods is aggregating arbitrary orientations text strings. Although this problem for neatly arranged text is usually seen as a solved problem, freestyle-arranged text still present a significant challenge because curvilinear text string and small gap between neighboring text strings present a challenge to aggregation algorithms developed for neatly arranged text string (see Fig. 1(b)).

* Corresponding author.
E-mail addresses: yzhang.user@gmail.com (Y. Zhang), stsljh@mail.sysu.edu.cn (J. Lai), pcyuen@comp.hkbu.edu.hk (P.C. Yuen).

http://dx.doi.org/10.1016/j.neucom.2015.05.028
0925-2312/

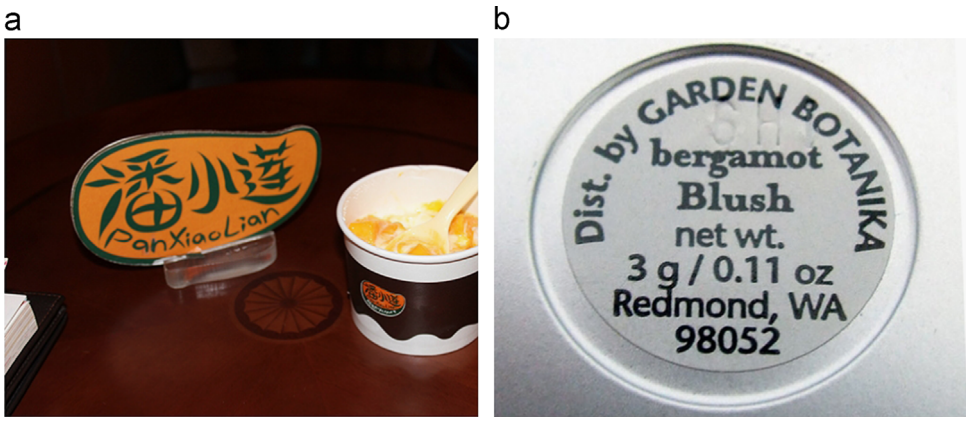

Fig. 1. Two major challenges of existing stroke-based text detection methods. (a) One challenge is extracting loosely constructed character, such as Chinese character. A Chinese character typically contains many CCs, whereas an English letter typically contains one CC (except lowercase "i" and "j"). (b) Another challenge is aggregating arbitrary orientations text strings.

Our goal is to develop a stroke-based method to detect arbitrary orientations text strings with loosely constructed characters in images. The proposed method consists of three steps. In the first step, the method employs the result of stroke width transform (SWT) as basic stroke candidates. The second step is merging these stroke candidates into characters using adaptive structuring elements. Finally, the individual characters are chained using *k*-nearest neighbors algorithm to identify arbitrary orientations text strings. Though widely used in the community, the ICDAR datasets [11,12] only contain horizontal English texts. In [5,6], two datasets with texts of different directions are released, but they include simple scenes without enough diversity in the languages and arrangements. Here we collect a new dataset with 400 images, which includes various characters and text strings in diverse scenes.

The main contributions and innovations of this paper are as follows:

- We propose a novel character extraction algorithm, in which the previous state-of-the-art work SWT is employed to extract possible text characters in the form of CCs in consistent stroke width, and dynamic structural element is defined for morphological processing to dilate the extracted text characters. The key idea of this operation is intended to solve detection problem of loosely constructed characters, which has not been particularly addressed in most of previous work.
- We investigate a novel method for text string aggregation, which is performed by searching *k*-nearest neighbors of each CC and building search path as a trace of text string in arbitrary directions. Text strings affected by curvature, inclination, and interleaving are robustly detected by the proposed aggregation algorithm.

The rest of the paper is organized as follows. We review related works in Section 2. The detail of the proposed approach is presented in Section 3. Experimental results and comparative studies are presented in Section 4. The paper is concluded with a discussion of future work in Section 5.

2. Related work

According to the features used and the ways they work, text detection approaches can be classified into two categories: texture-based methods and region-based methods.

Texture-based methods treat texts as a special type of texture and use distinct texture properties of text, such as local intensities, filter responses and wavelet coefficients. Some widely used features include local binary patterns (LBP), histograms of gradients (HOGs), Gabor filters and wavelets. These methods are computation demanding as all locations and scales are exhaustively scanned. Some machine learning methods are often used in this approach. Kim et al. [13] use a support vector machine (SVM) to analyze the textural properties of texts. The intensities of the raw pixels are fed directly to the SVM. No external feature extractor is required to reduce the dimensionality of the texture pattern, because SVMs work well even in high dimensional spaces. Yan et al. [14] present a text location method under complex background by combining Gabor filter and SVM. Four kinds of stroke features were extracted using Gabor filters, then the text location problem can be transformed into a texture classification problem, which can use SVM classifier for the purpose. Sin et al. [15] present a method of finding text regions using frequency-based features estimated. The Fourier spectrum is used to estimate the fundamental frequency of the text images. Zhong et al. [16] propose a caption localization method using discrete cosine transform (DCT) coefficients. The directionality and periodicity of local image blocks are used as texture measure to identify text regions. Li et al. [17] propose a text extraction scheme using key text points which can be acquired by the high-frequency sub-bands obtained from the wavelet transform. Shivakumara et al. [18] present a text detection method comprising of wavelet decomposition and color features. The wavelet decomposition is applied to obtain high-frequency sub-bands and then the average of the three sub-bands is computed further to enhance the text pixels. Texture-based methods are efficient in dealing with complex background. However, there is something unsatisfying about this kind of approaches, since the brute force nature of window classification is not particularly appealing and the computational complexity is proportional to the number of scales.

Region-based methods, on the other hand, first extract candidate text regions through edge detection [10], color clustering [6], or maximally stable extremal region (MSER) detection [19–21] and then eliminate non-text regions using various heuristic rules. Edge detection utilizes the geometry and structural properties of the character, since text regions contain a large number of edges. Connected component growing techniques, such as maximum difference (MD) [22] and morphological dilation, are used to group neighboring strokes with similar properties [23]. A typical example of region-based approaches was proposed by Epshtein et al. [7], where the stroke width transform is used to find the value of stroke width for each image pixel and non-characters was removed by a series of rules. Yao et al. [5] adopt SWT and also design various features that are intrinsic to texts and robust to