Text detection in natural scenes with multilingual text

Mikhail Zarechensky Scientific supervisor: Ph.D. Natalia Vassilieva

Department of Analytical Information Systems, Saint Petersburg State University

Abstract

Detecting text in natural scenes is an important prerequisite for further text recognition and other image analysis tasks. Most of text detection methods for scene images usually use a priori knowledge of language to detect text. As a rule such algorithms are evaluated on datasets which contain scenes only with text in English. This paper discusses known text detection algorithms and investigates them for invariance to the language.

1 Introduction

Recent advances in digital technology allow to take pictures from a large number of mobile devices. As a result, the number of photos taken by users is increasing every day. At the same time, we often have no annotations for images except those made by the device. Text in images provides important information about semantics of the image. Annotated images can be used in various applications, such as content-based image retrieval, automatic navigation, automatic translation. It is often the case that a language of a text in an image is not known in advance, or a single image contains text areas with text in different languages. How to effectively detect and recognize text in scene images is an actual research question. Text detection is an important prerequisite for further text recognition. In this paper we explore the problem of text detection.

In this paper, we discuss several known text detection algorithms and investigate them for invariance to a language. Quality of the text detection algorithm greatly depends on the shooting conditions and noise on the image, but in this paper we focus on the problem of language invariance of the algorithms in good conditions. First, we distinguish the main common steps of these algorithms. Second, we provide a theoretical estimation of language invariance for every step of the algorithms. Third, we perform experiments with two algorithms on different datasets to confirm theoretical result.

2 Related work

In order to recognize text in an image, it first has to be robustly detected. Unlike text detection for document images, text detection for scenes is still a challenging task due to the large variety of text appearance in images.

Proceedings of the Tenth Spring Researcher's Colloquium on Database and Information Systems, Veliky Novgorod, Russia, 2014

Figure 1: Sample images with multilingual text

Text in scenes can have different variations of the font style, size, distortion; it can have different contrast due to different lighting conditions. The whole image can also vary greatly. We should take into account low resolution, low contrast, heterogeneous background. Such variety gives rise to various approaches to text detection.

Existing methods for scene text detection can be broadly categorized into three groups: texture-based methods, region-based methods and hybrid methods.

Texture-based methods extract textural features of an image and then use machine learning techniques to identify text regions. It is common to extract textural features of image sub-regions using a sliding window and later classify every subregion as text or non-text. Thus these methods tend to be slow, because an image has to be processed at several scales. Another problem is construction of a whole text area from the coordinates of image sub-regions classified as text. Also image quality affects greatly these methods. Therefore, these approaches are difficult to use on mobile devices.

Region-based methods commonly use connected components labeling to extract components, which are character candidates. Next, various heuristics are applied to filter out non-character components. Remaining character candidates are grouped together to form text areas. Usually components are grouped based on their geometric properties. After character candidates are grouped into text, there may be additional checks to remove false positives. This approach is the most common. Performance of these methods mostly depends on heuristics to filter out regions that do not contain text.

Hybrid methods exploit region detector to detect text candidates and then segment image to extract character candidates. After character candidates are extracted, various heuristics are applied to eliminate non-characters as in the connected components based methods. Lastly character candidates are grouped into text.

In this paper we consider only connected components based methods. According to the results of the competition at the ICDAR 2013, this approach proved itself to be more effective comparing to others. We picked methods proposed by Yin et al. [12], Gomez et al. [5] and Chen et al. [2] for further consideration. These algorithms have

good results on the ICDAR datasets and use different approaches to detect text. All of these methods use the MSER algorithm for extraction of character candidates, so it is important to describe this algorithm in details.

The MSER algorithm

Maximally Stable Extremal Region (MSER) algorithm [7] is used for detecting character candidates in many state-of-the-art text detection algorithms [12], [2], [5], [11].

The input of the MSER algorithm is a grayscale image I. The output of the algorithm is a sequence of images $(I_t)_{t=0}^{255}$ which is created as follows. An input image I is successively binarized with a threshold t iterating from 0 to 255. The first image in the sequence I_0 is completely black. In the next images of the sequence white areas appear and grow. And the latest image I_{255} is completely white. There are also implementations of this algorithm when a sequence is constructed conversely. So the first image in the sequence is white and the latest image is completely black. White areas in the sequence are called extremal regions. For every extremal region it can be found for how many successive images in the sequence this region stays the same. Thus, by selecting a threshold value R, we can choose regions which are exactly the same in at least R successive images of the sequence. Such regions are called Maximally Stable Extremal Regions.

An advantage of the MSER algorithm is that it is well applicable for finding text character candidates. The MSER algorithm is invariant to affine transform, it can be applied to images with low quality, it has an efficient implementation. For example, the original implementation proposed by Matas et al. has the complexity of $O(nlog(log(n)))$, where n – is a number of pixels in the image. In particular, it is important that this algorithm is invariant to a language of text in images.

A disadvantage of the MSER is that it detects a lot of false positives – regions that do not contain characters. Therefore, it is necessary to apply additional checks to eliminate non-text regions. Also the MSER is quite sensitive to image blur. In case of a blurred image, some character regions may not be separable from each other.

Overview of text detection algorithms

The algorithm proposed by Yin et al. [12] was presented at ICDAR 2013 and got the first place in "Multi-script Robust Reading Competition in ICDAR 2013" [6]. It uses an approach based on the MSER algorithm to find character glyphs.

As we mentioned before, it is possible to detect character regions with original MSER algorithm even when an image is of poor quality. However, in this case a number of false positive character regions can be large. To solve this problem, the algorithm by Yin et al. [12] additionally performs parent-children elimination for the MSER tree. It improves accuracy of finding character regions. The main idea is to eliminate regions with very small or very big aspect ratio. That is, if at some moment of execution of the MSER algorithm, an extremal region violates the aspect ratio, then this region is removed from character candidates and is not processed further.

The next step of the algorithm is to group characters in order to construct text candidates. Character candidates are clustered into text candidates by the single-link clustering algorithm. Parameters for clustering algorithm – a distance function and a threshold are learned simultaneously using the algorithm called "self-training distance metric learning" which is also proposed by the Yin et al. The parameters depend on the following features: spatial distance, a differences between width and height, top and bottom alignments, color difference, stroke width difference. At the final step text candidates are labeled by a classifier as text or non-text areas. The following features are used to train the classifier: smoothness, the average stroke width, stroke width variation, height, width, and aspect ratio.

The second algorithm which we selected for analysis is the algorithm proposed by Gomez et al. [5]. This algorithm was presented at ICDAR 2013 and got second place in "Multi-script Robust Reading Competition in ICDAR 2013" [6].
This algorithm is a region-based algorithm and it uses the MSER algorithm at the first step for detecting text characters. The regions produced by the MSER are then filtered by the following features: size, aspect ratio, stroke width variance, and number of holes.

Next, a number of possible grouping hypotheses is created. The hypotheses differ one from another by image features. Then, these groups are analyzed based on the theory of Gestalt, formalized in [3], and only the most meaningful ones are kept. To construct the groups the following features are used: geometrical features, intensity and color means of the region, intensity and color means of the outer boundary, stroke width, gradient magnitude mean on the border.

To construct text candidates the single-link clustering algorithm is used with similar features.

At the final step a classifier is used in order to filter non-text candidates. To train the classifier the following features are used: stroke width, area, perimeter, number and area of holes.

Let us discuss an algorithm proposed by Chent et al. [2]. It uses a combination of the MSER and Canny edge detector [1] for detecting text candidates. In case when image is blurred this combination copes well because close symbols will be distinguished by Canny detector. This achieved by removing the MSER pixels outside the boundary formed by the Canny edges.

To filter out non-text regions the following features are used: size, aspect ratio, number of holes, stroke width.

For text candidates construction the single-link clustering algorithm is used, as the main parameters are spatial distance, width and height, aspect ratio. There is an additional check after text candidates are built. A text line is rejected if a significant portion of the objects are repetitive.

At the final step text lines are split into individual words using Otsu's method [10].

The main common steps of text detection algorithms

In general, by analyzing some of the most efficient algorithms on the ICDAR datasets, we can distinguish the main steps which are common for every algorithm:

1. Region decomposition: text character candidates extraction
2. Filtering regions using different heuristics to eliminate non-text candidates
3. Text line formation

3 Analysis of the main steps of the algorithms

In this section we discuss the main steps of the methods and provide a theoretical estimation of their language invariance.

Character candidates extraction

As presented above, for the region decomposition it is common to use the MSER algorithm. The MSER algorithm depends only on the intensity of the image. Since the text in the image tends to have equal intensity, at least in each symbol, the result of the algorithm is independent of language.

We can conclude that the MSER algorithm is equally applicable for region decomposition as for images, containing only one language and for images with multilingual text. As the Canny edge detector is not depend on a language, it follows that the modified MSER, proposed by Chen et al. is also invariance to a language

Filtering of regions

Let us review every feature used for region filtering.

- Aspect ratio

  Most letters of English language have aspect ratio being close to 1, so this feature might be useful to filter out false character candidates. To cope with elongated letters such as 'i' or 'l', a threshold should be small enough. On the one hand, this feature can be used for many languages because even if a letter has a very small aspect ratio and is filtered out, the absence of this letter will not affect the grouping of whole word at the grouping stage.

  On the other hand, when an entire word is not split into the characters it might cause difficulties in text detection. There are languages in which every word is continuously connected. For instance, Hindi, in which all words are linked by continuous line. In this case, rational use of this feature is difficult, because words might be very long. Thus this feature has limitations and may not be used for all languages.

- Region height

  Irrespective of language, height of the characters in one word are always about the same. Therefore, this type of filter is invariant to the language.

- Number of holes

  Number of holes in the English characters and in the hieroglyphs might be different. Therefore, this feature requires an additional configuration for different languages.

- Stroke width

  This feature is very important as it is shown in the work Epshtein et al. [4]. However, the proposed implementation has a limitation for the elements that have non-parallel edges. This feature of the implementation is essential for such languages as Arabic. Also the style of writing in Arabic language tends to have more variation in the stroke width, thus to achieve maximum efficiency, this feature must be configured for different languages separately.

Text line formation

Typically, to construct text one of the two approaches is used: methods based on machine learning and methods based on pairing the connected components using rules. Algorithms that use rules for pairing regions are quite stable for different languages because the main criteria for regions combination are spatial distance, lower and upper alignment. This features are invariant to a language, so all pros and cons of this approach will stay irrespective to a language.

At the same time, as shown above, algorithms that use machine learning techniques are using single-link clustering algorithm, i.e., distance between two clusters defined as the distance between two closest members of these clusters. Usually, the problem is to determine the distance function. For example, in the algorithm proposed by Yin et al.the distance function is a weighted sum of features where weight of each feature is determined by machine-learning techniques. To use this approach, you must have good training set. Other problem related to machine learning algorithms is overfitting, especially if there is a need to build a training set for many languages.

In order to emphasize the need of additional configuration for algorithms that perform characters grouping, it is enough to take into account Chinese characters consisting of several parts. In this case, not only characters must be grouped but also different elements of the same character. Therefore, weights of some features such as upper and lower alignment, width and height of the character, relative size, should not be big. Otherwise the probability to get an error of the second kind is increased because parts of the character will be interpreted by the algorithm as individual characters.

4 Empirical analysis

To confirm the theoretical estimations provided in the previous section we will perform a series of experiments for the two methods described in the section 2.

Descriptioin of the experiments

For the experiments two algorithms were selected: the one proposed by Yin et al. [12], and the algorithm by

Chen et al. [2] Let us remind, that to filter out non-text regions the first algorithm uses a machine-learning technique and the second one uses a rule-based approach. For evaluation we used a similar approach and the same quality measures as in the evaluation scheme of ICDAR 2013 competition. The following quality measures are used: precision, recall and f measure. They are defined as following: $recall = \frac{\sum_{i=1}^{|G|} match_G(G_i)}{|G|}$, $precision = \frac{\sum_{j=1}^{|D|} match_D(D_i)}{|D|}$, $f = 2\frac{recall \cdot precision}{recall + precision}$, where G is the set of groundtruth rectangles and D is the set of estimated rectangles. The matching functions are defined as following:

$$match_G(G_i) = \max_{j=1..|D|} \frac{2 \cdot area(G_i \cap D_j)}{area(G_i) + area(D_j)} \quad (1)$$

$$match_D(D_j) = \max_{i=1..|G|} \frac{2 \cdot area(D_j \cap G_i)}{area(D_j) + area(G_i)} \quad (2)$$

Description of test data

In the first group of tests we run the selected algorithms on the following datasets: MSRA-TD500, ICDAR 2011, ICDAR 2013. ICDAR 2011 dataset contains images with text in English only. MSRA-TD500 dataset contains images with text in English and Chinese. And ICDAR 2013 dataset contains images with multilingual text including Indo-Aryan languages and Chinese writing.

Also we created a new dataset which contains images with multilingual text. We included images in this dataset which are worse suited for text detection with heuristics as in algorithms of authors Yin et al. and Chen et al.. The majority of images of this dataset contain text in Hindi or in Arabic with a strong variability of stroke width, or contain text in Chinese where letters consist of several parts.

Figure 2: Sample images from special dataset

Analysis of the experimental results

The results of every test are presented in the following tables.

Based on the experimental results one may see that the difference between the result on the ICDAR 2011 dataset which contains only images with English text, and all others, is quite big. The minimum recall is reached on our special dataset, as it was expected.

5 Conclusion

In this work the following results were obtained.

- The most efficient text detection algorithms are discussed.
- The main common steps of text detection algorithms are identified.
- Every step of text detection algorithms is analyzed analytically for invariance to a language.

Table 1: Results on ICDAR 2011 dataset

Methods	recall	precision	f-measure
Yin et al.	0.68	0.86	0.76
Chen et al.	0.60	0.73	0.66

Table 2: Results on MSRA-TD500 dataset

Methods	recall	precision	f-measure
Yin et al.	0.21	0.517	0.335

Table 3: Results on ICDAR 2013 dataset

Methods	recall	precision	f-measure
Yin et al.	0.42	0.64	0.51

Table 4: Results on special dataset

Methods	recall	precision	f-measure
Yin et al.	0.079	0.577	0.109
Chen et al.	0.071	0.427	0.299

- Evaluated a series of experiments

During the work it was obtained that the existing set of features may strongly depend on a language. By changing settings of the rules that are used in the algorithms you can improve the text detection results on some predefined languages.

As a possible continuation to this work it is planned to implement a complete algorithm that solves the problem of text detection irrespective of a language. The analysis presented in this paper helps to identify problem pieces of the existing algorithms. The created dataset and the experimental results will allow to evaluate better the result of this new algorithm.

References

[1] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8:679–698, 1986.

[2] H. Chen, S.S. Tsai, G.Schroth, David M. Chen, R. Grzeszczuk, and B. Girod. Robust text detection in natural images with edge-enhanced maximally stable extremal regions. *IEEE International Conference on Image Processing*, Sep 2011.

[3] A. Desolneux, L. Moisan, and J.-M. Morel. A grouping principle and four applications. *IEEE Trans. PAMI*, 2003.

[4] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *CVPR*, 2010.

[5] L. Gomez and D. Karatzas. Multi-script text extraction from natural scenes. *ICDAR*, 2013.

[6] D. Kumar, M.N. Anil Prasad, and A.G. Ramakrishnan. Multi-script robust reading competition in icdar 2013. In *ACM Proc. International Workshop on Multilingual OCR, (MOCR 2013)*, 2013.

[7] J. Matas, O. Chum, M. Urban, and T. Pajdl. Robust wide baseline stereo from maximally stable extremal regions. In *British Machine Vision Conference*, volume 1, pages 384–393, 2002.

[8] S. Milyaev, O. Barinova, and T. Novikova. Image binarization for end-to-end text understanding in natural images. *ICDAR*, 2013.

[9] L. Neumann and J. Matas. Real-time scene text localization and recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[10] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

[11] I. Zeki Yalniz, Douglas Gray, and R. Manhmatha. Adaptive exploration of text regions in natural scene images. *ICDAR*, 2013.

[12] X.-C. Yin, X. Yin, and K. Huang. Robust text detection in natural scene images. *CoRR*, abs/1301.2628, 2013.