Detection and Recognition of Text in Natural Images using Feature Detection and Stroke Width Variation.

[1]Sathya Sri Pasham,
[1]Department of Information Technology,
Chaitanya Bharathi Institute of Technology, Hyderabad, India.

Abstract : The detection and recognition of text in natural images is used in modern software systems to perform tasks such as the detection of landmarks in images, mobile applications for the translation of visual signs, surveillance and much more. To detect and accurately identify text in natural images, a reliable and reliable method for distinguishing text from regions with no text in images is required. For this purpose, the extremely stable extreme regions algorithm (MSER), different geometric properties such as proportions, eccentricity, strength, extension, euler number, etc. and the stroke width variation algorithm should be used. Finally, the character recognition algorithm (OCR) is used to recognize previously detected text regions.

IndexTerms **-— MSER, Stoke width variation , Optical Character Recognition.**

I. INTRODUCTION

Detecting text in natural images, compared to scans of printed pages, faxes and business cards, is an important step for several Computer Vision applications, such as computer assistance for people with visual impairments, automatic business geocoding and robotic navigation in urban environments . The retrieval of texts both inside and outside provides contextual clues for a wide variety of viewing tasks. Furthermore, it has been shown that the performance of image recovery algorithms depends critically on the performance of their text detection modules. For example, two covers of books of similar design but with different text, are practically indistinguishable without optical detection and recognition of the characters of the text. When applied to images of natural scenes, the success rates of optical character recognition (OCR) decrease drastically. There are several reasons for this. First, most OCR engines are designed for scanned text and therefore depend on the segmentation that correctly separates text from the background pixels. While this is generally simple for the scanned text, it is much more difficult in natural images. Secondly, natural images show a wide range of image conditions, such as color noise, confusion, occlusions, etc. Finally, although the design of the pages for traditional OCR is simple and structured, in natural images it is much more difficult, because there is much less text, and there is a less general structure with great variability in both geometry and appearance. . Therefore, searching for text regions that use feature detection and the separation of this text from other elements of an image plays a fundamental role

II. METHODS AND PROCEDURES

The process of detection and recognition of text in natural images or unstructured scenes that contain undetermined or random scenarios is done initially by finding text regions in the image using Maximally Stable External Regions(MSER) feature detector. The MSER feature detector works well for finding text regions because the consistent color and high contrast of text leads to stable intensity profiles. Next ,the non text regions are removed using either basic geometric properties like aspect ratio , eccentricity ,euller ratio, extent, solidity etc. or by using stroke width variation . Finally,all the detection results are composed of individual text characters which are merged together into words or text lines for recognition tasks such as OCR.

2.1Text Detection using MSER

MSER (extreme extremely stable regions) is a method for the detection of blobs in images. The MSER algorithm extracts several covariant regions, called MSERs, from one image: an MSER is a connected, stable component of some gray level sets in the image. MSER is based on the idea of taking regions that remain almost the same across a wide range of thresholds. All pixels below a given threshold are white and all those above or equal are black. If we are shown a sequence of images with threshold I t with the t frame corresponding to the threshold t, we would first see a black image, then white spots will appear corresponding to the local intensity minima and then they will grow. These white dots will eventually merge, until the entire image is white. The set of all the components connected in the sequence is the set of all the extreme regions. The word extremal refers to the property that all pixels within the MSER have higher intensity (bright extreme regions) or lower (extreme dark regions) than all pixels at their outer limit

The extraction of MSER is implemented through the following steps. To begin, move the intensity threshold from black to white, creating a simple luminance threshold of the image. Then, extract the connected components, ie the extreme regions. The extreme regions have important properties, that everything is closed under the continuous (and therefore projective) transformation of the image coordinates, ie it is an invariant constant and it does not matter if the image is distorted or distorted. It continues to find a threshold when an extreme region is maximum stable, that is a local minimum of the relative growth of its square. Due to the discrete nature of the image, the under / above region may coincide with the current region, in which case the region is still considered to be the maximum.

Finally, approximate a region with an ellipse (this step may be optional) and retain the descriptions of those regions as characteristics. MSER works well on images that contain homogeneous regions with distinctive boundaries. It is particularly cost-effective for small regions while it does not work well with images with any motion blur. The MSER regions detected in the natural image are shown in Figure 1 below.

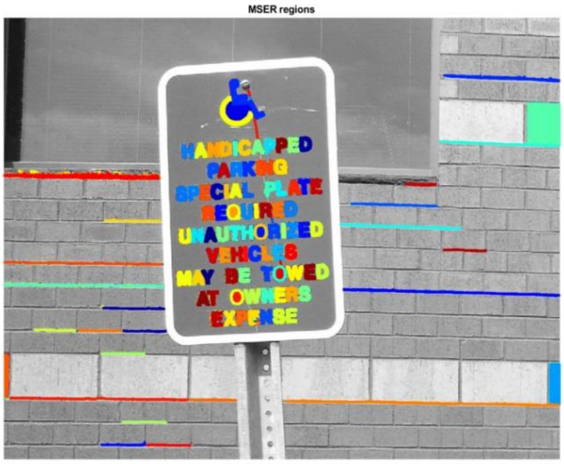

Figure 1 MSER regions detected in a natural image

2.2 Filtering Non Text Regions

Although the MSER algorithm selects most of the text, it also detects many other stable regions in the image that are not text. A rule-based approach can be used to eliminate non-text regions. For example, the geometric properties of the text can be used to filter regions without text using simple thresholds. Once the regions of connected components [3] (MSER) have been detected, we can use certain discriminant and geometric properties to filter candidates from regions other than text. Regions are filtered through the use of thresholds. Property values are characteristics by which thresholds can be established, calculated or learned manually using machine learning techniques. These features are a quick and easy way to distinguish non-text regions from text features in images [2]. Below is a brief discussion of the discriminatory and geometric properties used in the system

- Aspect ratio: the relationship between width and height of the bounding boxes.
- Eccentricity: used to measure the circular nature of a given region. It is defined as the distance between the foci and / or the ellipse and its major axis.
- Solidity: it is the ratio between the pixels in the convex area of the hull that are also found in a given region [2]. It is calculated per convex area / area.
- Extension: the size and position of the rectangle that encloses the text.
- Euler Number: it is a characteristic of a binary image. It is the number of connected components minus the number of holes (in those components)

2.3 Stroke Width Variation

Stroke width is another property that is used to distinguish (and filter) non-text regions from text areas. Epshtein et al [1] introduced the approach to the use of stroke width to help detect regions of text in images. The main idea behind the use of stroke width to filter the region without text is based on the fact that, in general, the text compared to other elements in an image has a constant stroke width. This means that binary images can be transformed into stroke width images (skeletal images) and these skeletal images are used to calculate the variation of the stroke width. This relationship can be used with a maximum variance threshold to filter regions without text. Because stroke widths are calculated on a per pixel basis, text detection systems that use this technique can detect text in a manner that is not sensitive to character, color, size, language, and orientation. Initially, we used the Stroke Width transformation to group the pixels into candidate letters and then we started the mechanism to group the letters into larger words and lines constructions that allow more filtering. The flowchart of the algorithm used in Stroke Width Variation is shown in figure 2

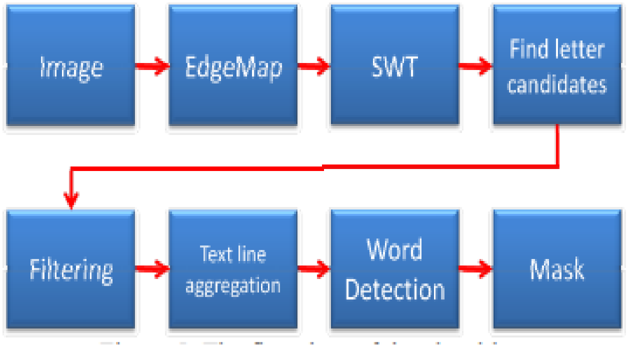

Figure 2 2 Flowchart of the Stoke width variation algorithm

2.3.1 Stoke width transform

The stroke width transformation is a local image operator that calculates the width of the most likely stroke that contains the pixel by pixel. The output of the Stoke width transformation is an image that is equal to the size of the input image where each element contains the width of the stroke associated with the pixel. We define the stroke as an adjacent part of an image that forms a band of almost constant width. We do not assume that we know the true width of the stroke, but that we recover it.

2.3.2 Search by letter

The output of the stroke width transformation is an image in which each pixel contains the width of the most probable line to which it belongs. You can group two adjacent pixels together if they have a similar stroke width. Thus, components that may contain text are identified, so a small set of fairly flexible rules is used. The first test we perform is to calculate the variance of the stroke width within each connected component and reject those of which the variance is too big. This rejects areas like foliage, which prevails in many natural images and is known to be difficult to distinguish from the text. Many natural processes can generate long and narrow components that can be confused with possible letters. Further rules these components, limiting their aspect ratio to a value between 0.1 and 10. Similarly, we limit the relation between the diameter of the connected component and its width of the median stroke to a value lower than 10. Finally, components whose size is too small or too large can be ignored and the remaining components are considered candidates for letters

2.3.3Grouping of letters in lines of text

The search for groups of letters is a significant filtering mechanism, since loose letters do not usually appear in images and this reasoning allows us to eliminate randomly dispersed noise. An important key to the text is that it appears in a linear form. The text on a line is expected to have similarities, including a stroke width, letter width, height, and a similar space between letters and words. Including this reasoning proves to be direct and precious. In addition, the average colors of the candidates are compared for matching, since it is normally expected that the letters in the same word are written in the same color. In the next phase of the algorithm, the candidate pairs determined above are grouped into strings. Initially, each string consists of a single pair of candidate letters. Two chains can join together if they share one end and have a similar address. The process ends when it is not possible to join the strings. Each string produced of sufficient length (at least 3 letters in our experiments) is considered a line of text.

Figure 3 shows the application of Stoke width variation on a natural image,that removes the non text regions in the image after MSER feature detection on the image

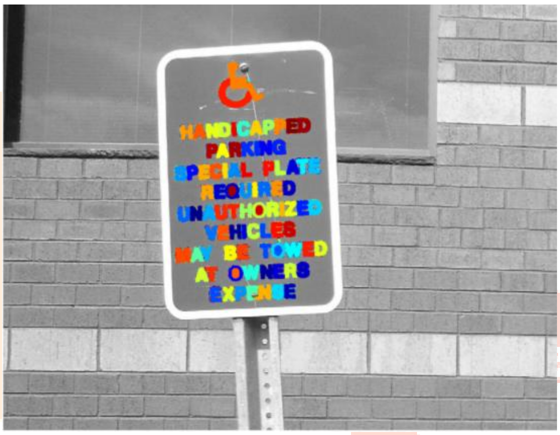

Figure 3 Image after Stoke Width Variation is applied

III. TEXT DETECTION USING OCR (OPTICAL CHARACTER RECOGNITION)

Detects the text in the image and passes the bounding boxes of the detected text to an ORC algorithm to determine the text that was found in the image. A typical way to create an OCR system is to first train a machine to get descriptions of how the characters appear, and then compare the unknown characters with the descriptions previously learned to get the best match. In a typical OCR business system, the training process was carried out in advance. The matching procedure generally consists of a pre-processing phase, a recognition phase and an optional post-processing phase [5]

In the preprocessing phase, the regions containing the text are found and each character is segmented by the word. Then, a leveling and normalization process is applied to eliminate noise and variation in size, inclination and rotation before performing the recognition. The methods of recognition consist in the extraction of characteristics and classification. The extraction of features aims to capture the essential characteristics of the characters. Most of the characteristic spaces are empirically designed based on experiments and domain knowledge. The fundamental idea is to make the characteristics invariable with global distortions, sources and deformations. The goal of the classification is to calculate the odds or scores that an unknown character belongs to each class of characters and label it as the class that has the highest probability or score. Common distance classifiers, such as the nearest neighbor, the Bayesian quadratic classifier and artificial neural networks are often used to perform this task. In the step after processing, the identified characters are grouped together to reconstruct the original words or numbers.

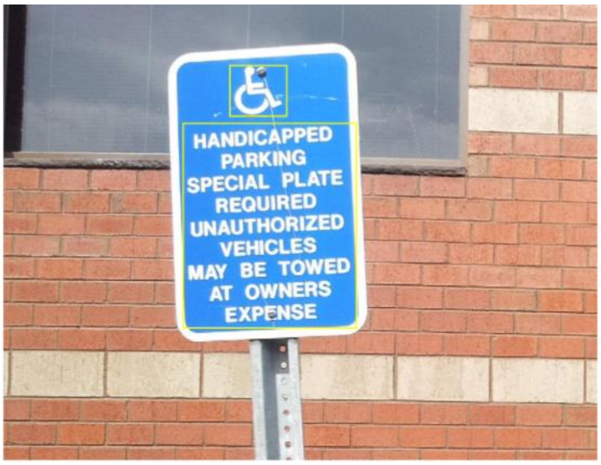

Figure 4 Shows the detected text in the natural image that is to be recognised by OCR

IV. CONCLUSION

In this article, solutions for automatic detection and recognition of text in natural images is proposed. For the text detection challenge, we pursue a detection via segmentation approach using MSER. Later filtering of non-text regions using geometric properties and Stroke Width Variation is implemented. In stroke width variation approach we define the notion of a stroke and derive an efficient algorithm to compute it, producing a new image feature. Unlike other features used for text detection, the proposed SWT combines dense estimation (computed at every pixel) with non-local scope (stroke width depends on information contained sometimes in very far apart pixels). There are several possible extensions for this work. The grouping of letters can be improved by considering the directions of the recovered strokes. This may allow the detection of curved text lines as well. For the text recognition task ,a typical commercial optical character recognition system is implemented.

REFERENCES

[1] E. Ofek B. Epshtein and Y. Wexler. “Detecting text in natural scenes with stroke width transform”. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, no. d, (2010), pp. 2963–2970

[2] L. M. Bergasa and J. J. Yebes. “Text Location in Complex Images”. In: International Conference on Pattern Recognition, no. Icpr (2012), pp. 617–620.

[3] L. Neumann and J. Matas. “Efficient Scene text localization and recognition with local character refinement”. In: Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 2015- Novem (2015), pp. 746–750.

[4] C. Republic. “Real-Time Scene Text Localization and Recognition”. In: IEEE Conference on Computer Vision and Pattern Recognition (2012).

[5] C. Y. Suen S. Mori and K. Yamamoto. “Historical review of ocr research and developement.” In: Proceedings of the IEEE (1992), pp. 1029–1058

[6] Teresa Nicole Brooks “Exploring Geometric Property Thresholds For Filtering Non-Text Regions In A Connected Component Based Text Detection Application”