OPTICAL CHARACTER RECOGNITION IN SCENE IMAGES

Xiaojie Jiang, William Makabenta

Department of Computing Sciences, Villanova University, Villanova, PA 19085, USA

ABSTRACT

Text detection in scene or natural images is an interesting problem. In this paper, we utilized the methods described in [1]. In particular, we applied Maximally Stable Extremal Regions to detect basic letter candidates. After finding the Maximally Stable Extremal Regions, the intersection between the regions and the Canny edges are used to find the connected component text candidates. These connected components are then filtered by their geometry and stroke width to eliminate non-text objects. Thereafter, we employed optical character recognition technique on the candidates for text recognition. We tested our implementation with a small set of images that was obtained online using Google. The result of our tests demonstrated acceptable performance for the constraints we imposed.

Index Terms— Text detection, maximally stable extremal regions, optical character recognition

1. INTRODUCTION

In today's world, mobile devices are dominating the web. With the increasing amount of mobile users, the amount of images that have surfaced the web has definitely increased. In is particularly interesting to see that most technology goliaths has some sort of image recognition tools, such as Google's image search and Amazon's product search. In these tools, local image features are extracted from images taken by the user, and then being matched with a large database. Although these tools are working very nicely, and have achieved great results, they are ignoring one specific class of features in images: text.

In order to extract text from an image, the text within the image must first be located. However, detecting text in an image is a challenging task due to the fact that features, such as the size, fronts, color, stroke widths, and distortions, of the text can vary greatly from image to image.

In our implementation, we first employed Maximally Stable Extremal Regions (MSER) on the original image to locate the possible text candidates. We also applied Canny edge detection to the image to account for the MSER method's sensitivity to blurred images and small text located in images of limited resolution. The complimentary properties of Canny edges detection and MSER are combined to give us better results. Further, we filtered the connected components found in the edge-enhanced MSER by common text geometric properties and stroke width in order to remove non-text regions. Finally, the remaining text candidate regions separated by their bounding boxes and were then passed to OCR for text recognition.

The remainder of this paper is organized as follows. Section 2 we provided some background information. Section 3 we described our methodology. Section 4 we discussed our experimental results and Section 5 concludes the paper.

2. BACKGROUND

Our implementation depends largely on the edge-enhanced MSER and filtering out non-text locations (Section 2.1). For filtering, our implementation employed geometric filtering (Section 2.2) and stroke width filtering (Section 2.3).

2.1. Edge-enhanced MSER

Maximal Stable Extremal Regions are used as a method of blob detection – blob detection refers to detecting regions in an image that differs in properties, such as brightness or color. These blobs are typically found by searching for regions that maintain a consistent intensity when a wide range of thresholds is applied to an image. As the contrast of text within an image with its background is typically significant or the color of every letters in the text stays consistent, MSER is a great choice for text detection. However, MSER is sensitive to image blur, [1] suggested enhancing the text candidate by applying Canny edge detection on the original image. By finding the intersection of MSER and Canny edges, we combined the complimentary properties of Canny edge detection and MSER region to achieve a better result. Fig 1 shows an image after applying edge-enhanced MSER.

Fig. 1. An image after applying edge-enhanced MSER

Then the edges are grown along the gradient and removed along the gradient in order to remove some of the non-text pixels in the MSER and edge intersection. The result of growing and removing the edges along the gradient are more segmented regions.

2.2. Geometric Filtering

The result of edge-enhanced MSER is a binary image where the foreground connected components are considered as letter candidates. We perform a set of simple geometric checks on each connected components to filter out non-text objects. First, very large and very small objects are rejected. We assume that the size of text within an image does not take up the whole image nor does it only make up a very small portion of the image. Then, since most letters have aspect ratio close to 1 [1], we reject connected components with very large and very small aspect ratio. Finally, we eliminate components that contain a large number of holes, because letters does not have many holes. Fig.2 shows the result of geometric filtering. Almost all of the non-text from edge-enhanced MSER objects had been eliminated.

Fig. 2. Result of geometric filtering

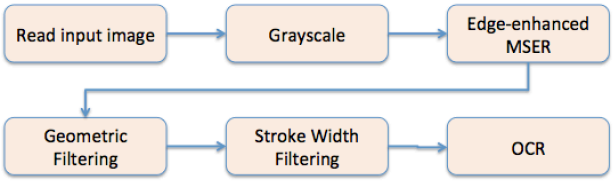

Fig. 3. Implementation flowchart

The three main region properties that are used to filter the connected components by the typical text attributes are the eccentricity, area, and solidity of the object. MSER regions can usually be drawn with a surrounding ellipse, so the eccentricity of the ellipse is used to determine the aspect ratio. The area, or the number of pixels in a region, is used to remove the very large and very small objects that are assumed to be non-text components. The solidity, or proportion of pixels in the convex hull that are also in the region, of letters is typically low, so the connected components with a high solidity can be rejected. The connected components that were removed tend to have an eccentricity greater than 0.995, an area less than 80 pixels or greater than 1000 pixels, and solidity under 0.4.

2.3. Stroke Width Filtering

Characters in text usually have a similar stroke width or thickness. This text property can be used to continue to filter out more non-text regions remaining after filtering the edge-enhanced MSER by the geometric properties of the connected components. In order to find the stroke width of each object, we use a distance transform approach. The distance image can be found by setting the intensity of each pixel within a text candidate region to the distance from that pixel to the nearest background pixel. The stroke width information can then be derived from the distance image. Since the stroke width of characters in a line of text are consistent among characters, regions with too much variations in object stroke width can be assumed to be non-text and removed from the candidate regions. A text candidate region is removed when the stroke width standard deviation divided by the stroke width mean is greater than 0.35 (std/mean > 0.35).

2.4. Text Region Bounding Box

Once the connected components are removed until remaining components are mostly text characters, the individual characters have to be merged into a single connected component. This is accomplished by performing morphological operations on the connected components until the outliers are rooted out. Morphological closing, which is essentially dilation followed by an erosion, is used to enlarge the boundaries of the foreground to shrink any small holes in the region. Then morphological opening is applied to the image by performing an erosion followed by a dilation. The opening removes any stray foreground pixels in the background. The bounding box is then found for each region with an area greater than a specified threshold (5000 pixels). Each of these bounded regions can then be passed through to an OCR system to recognize the text in each region.

3. METHODOLOGY

The flowchart of our implementation is shown in Fig.3. At the start of the application, the program will read in an input image. Following that the image will be converted to a gray-scale image. Once the image is converted, the program performs the MSER on the image and enhances the result by employing Canny edge detection on the gray-scale image. As a next step, the resulting connected components are filtered using geometric filtering on properties like numbers of holes and aspect ratios. The result of geometric filtering will be passed on for stroke width filtering, objects with high variation in stroke width are rejected. Then the text regions are located by using morphological closing and opening to clean up outliers and finding the bounding box for each text region. Finally, the image will be passed on to OCR for text recognition.

4. EXPERIMENTAL RESULTS

In order to evaluate our implementation, we apply it to several images we've found online via Google's image search.

Fig.4 shows the original images before employing the edge-enhanced MSER.

Fig.5 shows the images after employing edge-enhanced MSER. The separation between the text and the image seems acceptable.

Fig.6 shows the images after geometric filtering. As shown in the figure, some of the letter candidates are considered as non-text region, in particular the letter '3', also the stroke in the letter 'r' were erased. In the middle image, the smaller text at the bottom of the sign is considered too small and is rejected as a text candidate.

Fig.7 shows the images after stroke width filtering. As shown in the figure, more letter candidates were deleted, in particular the letter 'm' in museum.

Fig.8 shows the result of the OCR system provided in MATLAB. Due to the fact that many letter candidates were erased, the OCR was not able to detect all text in the images. However, for the remaining text candidate regions, OCR was able to provide acceptable results. The only error in the OCR was in 'Quiznos-Sub', the text recognition identified the letter 'b' as letter 'n'.

The reason that some connected components were being removed was because the implementation tried to work with all available images, and as mentioned earlier, the texts in images can vary greatly depending on it's font, stroke width, and size. The geometric filter value can also be adjusted to prevent the removal of real text characters from the letter candidates. However, we believe that our implementation gave us acceptable results.

Fig. 4. Original images

Fig. 5. Images after edge-enhanced MSER

Fig. 6. Images after geometric filtering

Fig. 7. Images after stroke width filtering

```
text2 =

A 9H
Equipment Poom
```

```
text3 =

Fa m
House
uscum
```

```
text4 =

Quiznos-Sun
```

Fig. 8. Result of OCR

5. CONCLUSION

In this paper we followed [1] proposal and implemented the system as described by employing Maximally Stable Extremal Regions along with Canny edge detection. Further more, we also utilized geometric filtering and stroke width filtering. The results of our experiment were considered as acceptable to us. However it does not seem promising. Nonetheless, we believe that with enough time and a much more thorough study in scene image recognition, this method can definitely be very useful.

6. REFERENCES

[1] H. Chen, S. S. Tsai, G Schroth, D. M. Chen, R Grzeszczuk, B Girod "Robust text detection in natural images with edge-enhanced maximally stable extremal regions," *Image Processing (ICIP) 2011*, IEEE, Brussels, pp. 2609 – 2612.