Text detection and classification in natural images

Andrei Bajenov (abajenov@stanford.edu), Haisam Islam (hislam@stanford.edu)

Introduction

Text detection in natural images is important for various applications in computer vision. Most Optical Character Recognition (OCR) algorithms, however, are designed for scanned text and perform poorly in natural scenes, where the text and background are not easily segmented, imaging conditions (amount of noise, blur, occlusions, etc.) vary, and there is less structure in the geometry and appearance of the text.

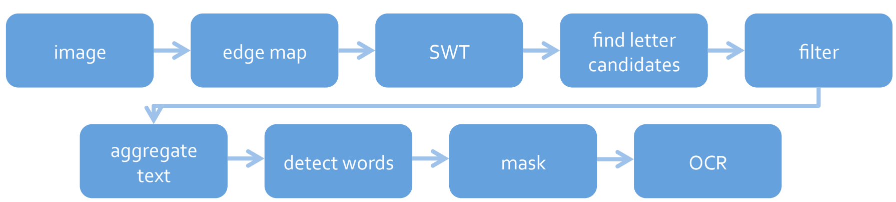
A recent approach developed by Epshtein *et al* utilizes the fact that text usually has slowly varying or constant stroke width [1]. The method uses a local image operator to compute stroke width and makes reasonable assumptions about the geometry of text to identify and group text together, and has shown greater performance than other text detection algorithms. Some advantages of this method include a substantial reduction in number of falsely detected pixels, absence of the need to scan the image at multiple scales, and detection of text at any orientation and of any language. The detected text can then be processed further before being run through an OCR algorithm.

Goals

We propose to implement the text detection algorithm based on the stroke width, and classify the detected text using the OCR algorithm Tesseract [2]. We will try to make these steps robust to different imaging conditions, and will explore various processing steps throughout the pipeline to optimize the performance of the text detection and classification, and will test on images of buildings found throughout the Stanford campus.

Approach

A stroke is defined as a continuous banded region. The stroke width at each pixel is calculated by the Stroke Width Transform (SWT), which requires an edge map. The pixels are then grouped into letter candidates, and a filtering process is applied. Letters are aggregated into groups to reduce single detections, which are uncommon and thus likely false. Finally, the text is separated into words, a mask is generated, and the output is run through an OCR algorithm. The processing pipeline, mostly taken from [1], is given below.



Alternative processing schemes may be used to increase the accuracy of the text detection, such as using Maximally Stable Extremal Regions to find initial letter candidates [3], or using *k*-means on feature vectors obtained from detected text to distinguish true text from false positives [4].

References

1. Epshtein, Boris, Eyal Ofek, and Yonatan Wexler. "Detecting text in natural scenes with stroke width transform." Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010.
2. Smith, Ray. "An Overview of the Tesseract OCR Engine." ICDAR. Vol. 7. No. 1. 2007.
3. Chen, Huizhong, et al. "Robust text detection in natural images with edge-enhanced maximally stable extremal regions." Image Processing (ICIP), 2011 18th IEEE International Conference on. IEEE, 2011.
4. Mosleh, Ali, Nizar Bouguila, and A. Ben Hamza. "Image Text Detection Using a Bandlet-Based Edge Detector and Stroke Width Transform." BMVC. 2012.

We will not use a DROID camera phone.