Automatic License Plate Detection

Guo Chengxi
The Chinese University of Hong Kong, Shenzhen
118010080@link.cuhk.edu.cn

Abstract

Automatic license plate recognition (ALPR) is an important technique in most of the traffic-related applications. The difficulty of this task lies in the non-uniformity of license plate and various illumination conditions, which is still being actively researched. In our work, a plate detection algorithm was implemented to obtain the license plate from the whole image. Our detection method is maily based on the extraction of edge information and the property of connected components. The experiment results showed that the recall rate of detected plates is satisfactory, and the precision rate is worse, where plate classifier is needed. Preprocessing is also conducted so that our result images can be applied to plate recognition tasks.

1 Introduction

Automatic license plate recognition (ALPR) is the extraction of vehicle license plate information from an image. Although currently assorted mature techniques regarding this issue have been proposed, there are still various problems with ALPR systems. Major challenges include the varying characteristics of the license plate (e.g. language, size, style) and unfavorable illumination conditions (which may be caused by factors such as weather, distance) [1]. Therefore, further research is needed in this area. In this paper, we implemented a license plate detection algorithm. The goal of the algorithm is to output the extracted license plate from input of original images. Our results can be directly fed as the input of end-to-end neural networks or perform single character recognition after segmentation.

1.1 Related Work

Existing license plate extraction methods can be divided into the following categories. The first is using boundary or edge features, based on the rectangular shape of license plate. Sobel filter is mainly used to detect edges, and two vertical lines of the license plate are considered as a robust feature since detecting horizontal lines will result in errors due to car bumpers [3]. This edge-based method is simple and starightforward, but can be hardly applied to complex scenarios since they are sensitive to unwanted edges. Another method is to utilize global image features to find connected object with plate shape. This method is independent of the license plate position, but may generate broken objects. Other strategies include using the features of texture or color, which will not be considered in our approach. Furthermore, hybrid methods have been used, which combine several features such as color and edge or frequency information to obtain optimal results.

2 The Proposed Algorithm

Our pipeline is summarized in Figure 1. The algorithm will be introduced in the following three sections: sobel filtering and morphological processing, contour extraction and floodfilling, image postprocessing.

Paper Report of Final Project for EIE4512 - Digital Image Processing.
The Chinese University of Hong Kong, Shenzhen.

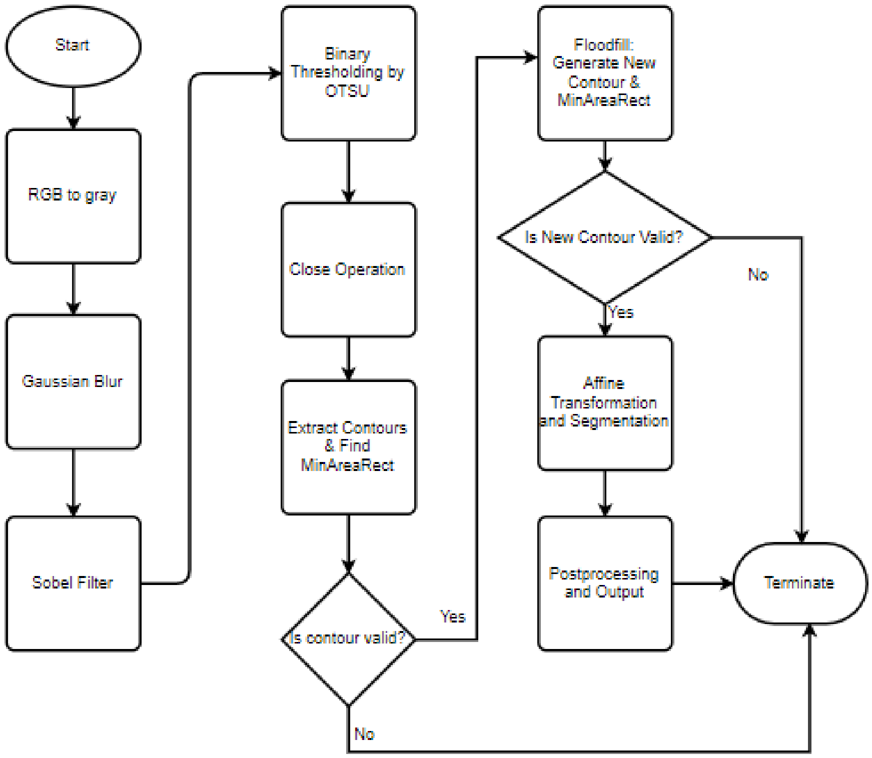


Figure 1: Pipeline of Our Algorithm

2.1 Sobel Filtering and Morphological Processing

To localize the rectangular license plate region, we are interested in the edge features of the image. Therefore, we choose sobel operator to extract the edge information. The image (of RGB format) is first converted into grayscale, and then smoothed by Gaussian lowpass filter. Next, sobel filter is passed to the image, highlighting the horizontal gradient magnitude. Added with binarization, the interested regions appear purely white in the dark background. The reason why we did not use other edge detectors (e.g. canny edge combined with hough transform) is that they will detect other irrelevant edges, which is unfavorable to locate the accurate enclosed region. Figure 2 shows the gaussian blur process and Figure 3 shows the sobel filtering process.

(a)

(b)

Figure 2: (a) original grayscale image; (b) image after Gaussian blur

With the small edges of characters detected, it is not enough to find the accurate boundary for plate. Therefore, we utilize close operation in morphological processing to connect the "white plate characters" and form compact plate region. As shown in Figure 4, the contour of the plate is clearly visible.

(a) (b)

Figure 3: (a) image after sobel filtering; (b) image after binarization

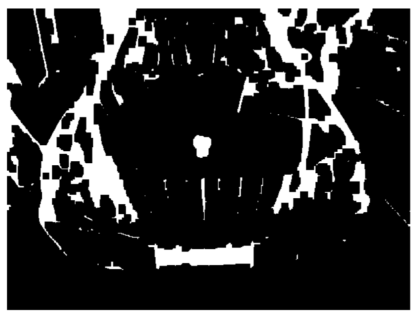


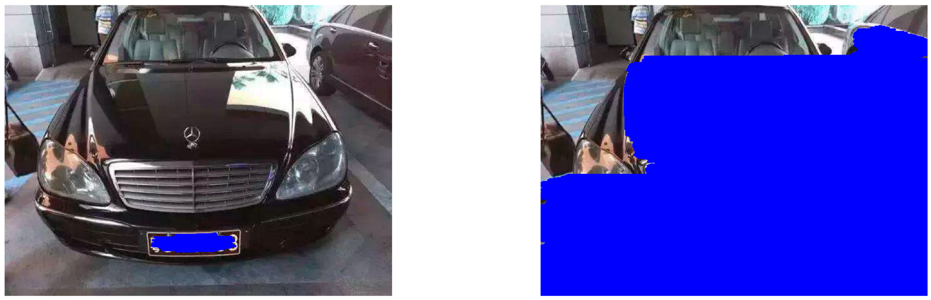
Figure 4: image after close operation

2.2 Contour Extraction and Floodfilling

In the next step, all the contours (boundary of connected region) in the image are extracted using opencv function. Since the contours are of irregular shape, the rectangles enclosing them of smalllest area are also found, which can accurately extract the plate boundary. But due to environmental noise, the extracted rectangles need to be verified. In our experiment, the area and the aspect ratio of the rectangle are considered. The plate should have relatively small variations on the width-to-height ratio, and the area also should not be too small or large, though fluctuations exist due to various shooting distances.

Figure 5: interested contours in the image (boundary in blue line)

However, after the above procedures, some irrelevant regions of similar sizes and shpaes will still be detected as plate. To reduce error, we propose to use the floodfill algorithm implemented in *opencv*. The basic principle is to generate random seeds in the specified rectangle region, and then the neighboring regions with close color will be covered with the seed's color. Since plate is a close rectangular region, it will detected as valid one and preserved, while other non-interested regions will be eliminated. As shown in Figure 6, the plate area is flooded with blue color in the left figure, but the right one, whose seeds are not in the plate region, is almost flooded with blue.



(a) (b)

Figure 6: (a) correct contour after floodfill; (b) incorrect contour after floodfill (caused by the right thin contour shown in Fig.5

2.3 Image Postprocessing

Figure 7: Localized Plate Boundary

Utilizing the above methods, the plate region is localized. To successfully obtain the plate, our next procedure is to perform affine transformation to the image, because the plate in images has certain rotated angle. Knowing the angle of the interested rectangle and its center's coordinates, we can derive the affine transformation matrix and place the plate horizontally. Then, Gaussian filter are passed to the image and histogram equalization is performed. Eventually, the outputs are obtained as shown in Figure 8, which can be further applied as inputs to plate recognition tasks.

(a) (b)

Figure 8: (a) extracted plate region; (b) plate image after equalization and grayscaling

3 Experiments and Discussions

3.1 Dataset

The dataset used in this project contains 47 car images which are partially collected from [3]. It is shooted in various environments with different illumination effects, and the position, size and angle of the plates vary depending on the shooting distances and perspectives. The vehicle plates are all Chinese for simplicity of the detection task, considering that the size and shape of plates differ in regions.

3.2 Evaluation Criteria

3.2.1 *Precision & Recall*

Two standard criteria will be considered when we evaluate the performance of our algorithm. The first one is precision and recall. Precision quantifies the number of positive class predictions that actually belong to the positive class, while recall computes the number of positive class predictions made out of all positive examples in the dataset. The calculation is as follows:

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

where TP is true positive, FP is false positive, and FN is false negative. Specifically in our task, true positives are the accurately detected plates, false positive is the irrelevant regions that are detected as plates, and false negatives are the plates not detected. In our experiment, the judgement of ground truth are based on visual inspection.

3.2.2 *Intersection over Union*

The second metric will use is intersection over union (IoU). It measures the degree of overlap between two specific regions. As its name suggests, it is the ratio of the area of intersection over the area of union. In our case, these two regions are the actual region of interest and the region detected by the algorithm. This measure is chosen because some detected plates may not be compact, which may be caused by the fragmented nature of detected region inside the plates.

3.3 Test Results

Some of the successful test results are shown in Fig. 9. The rectangle enclosed by red lines is the detected license plate region. The two measures are calculated and shown in Table 1.

(a) (b) (c)

Figure 9: Successful Test Results

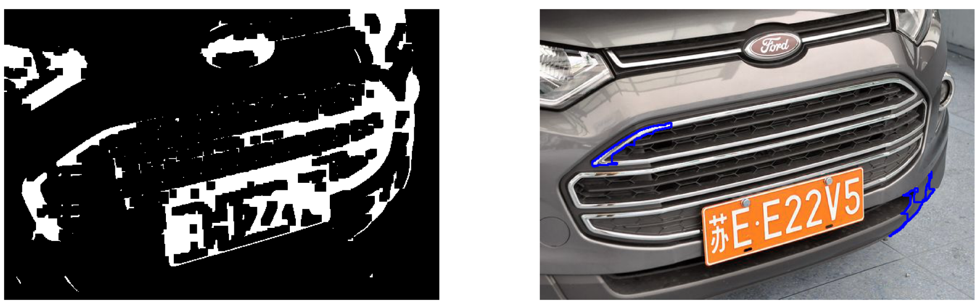
Table 1: Performance of Our Algorithm

Evaluation Metrics		
Precision	Recall	IoU
34%	76%	0.86

3.4 Discussion

As shown in Table 1, the recall rate of our algorithm is relatively satisfactory, while the precision is relatively low. This is caused by the irrelevant edge information that persists in the images. In images with complicated textures, other irrelevant edges will be detected, and if their shape and size are similar to the plate, they will pass our tests of valid rectangles. This shows that to achieve higher precision, information other than edge and connected region should be considered, such as character texture or color.

As for the recall rate, one major challenge is to correctly distinguish the high-frequency information in the image. Because in our datasets, the resolution of the images are not fixed, and may vary a lot. In smooth images, edges cannot be easily detected; while in image of high contrast and frequency, the edges are too closed so that irrelevant regions (false positives) will be wrongly connected and the extracted result will fail. One failure case is shown in Fig. 10. Since the edges are too close the plates, once the image performs closing operation, the plate region will be "contaminated"



(a) (b)

Figure 10: One failure case: the edge information interfere with the correct region detection. (a) image after closing operation; (b) detected contour

4 Citation

[1] Maglad, Khalid W. "A vehicle license plate detection and recognition system." Journal of Computer Science 8.3 (2012): 310.

[2] S. Wang and H. Lee, “Detection and recognition of license plate characters with different appearances,” in Proc. Int. Conf. Intell. Transp. Syst., vol. 2. 2003, pp. 979–984.

[3] Xu, Zhenbo, et al. "Towards end-to-end license plate detection and recognition: A large dataset and baseline." Proceedings of the European conference on computer vision (ECCV). 2018.