Automatic License Plate Localization using Eight Neighbourhood Connected Components Labelling

Muhammad Fikri Izzuddin bin Rosli
Faculty of Information Science and Technology
Universiti Kebangsaan Malaysia
Bandar Baru Bangi, Selangor, Malaysia
a162027@siswa.ukm.edu.my

***Index Terms*—license plate, number plate, localization, computer vision**

I. Introduction

The number of registered vehicles are increasing at rapid rate due to growth of economics and improvement of living standard. Toll payment, surveillance system and traffic camera are some of the examples license plate recognition. Due to increased number of vehicles, this processes has become impractical for human to manually identify each vehicle license plate. This lead to the automation of license plate recognition. License plate, or number plate is defined as a rectangular plate located at front and rear of a vehicle for the purpose of identification. The color and characters of the license plate vary across country. For example, in Europe the left region is for the country code followed by unique alphanumeric text that identifies the vehicle.

II. Problem Statement

Even though human intuitively understand what is a license plate, computer does not have the intuition that human has. Hence, a more specific and detailed definition of license plate is required in order for computer to 'understand' what is a license plate. This complexity of the problem increased because each country has different style of license plate. For this project, only the license plate from European Union are considered.

III. Objectives

To identify the location of license plate in the image using morphological operations and Eight Neighbourhood Connected Components Labelling (CCL).

IV. Methods

A. Image Acquisition

In this project, dataset from Ondrej Martinsky's license plate dataset. The dataset contains 97 images of front and rear views of various vehicles. The design and pattern of license plate are commonly found in the European Union (EU). The background color of the number plate is white while the color of the characters are black. The dataset is split into 30 images for training set and the remaining 67 images for test set. Figure 1 show example of input image in RGB.

Fig. 1: RGB Image

B. Color Conversion

First of all, convert the RGB color images to grayscale. This step is to reduce the complexity of the problem by reducing from 3 channel to 1 channel. Equation 1 was used to convert the RGB image to grayscale. This equation favor green channel as it is the color that most sensitive to human eye. Figure 2 show the result of color conversion.

$$I_{gs} = 0.299 * R + 0.587 * G + 0.114 * B \tag{1}$$

where I_{gs} is grayscale image, R is red channel, G is green channel and B is blue channel.

C. Sharpening

Sharpening was used to increase the contrast of the image. This process is important as it help to improve the performance of the license plate localization. The equation 2 was used to apply the histogram equalization. Figure 3 is the output of the image sharpening.

$$f_{eq()} = \left[H(a) \times \frac{K-1}{WH} \right] \tag{2}$$

where $f_{eq()}$ is the frequency of the equalization, a is cumulative position, K is the intensity value, W is the image width and H is the image height. In this project the intensity value, K is set to 256.

Fig. 2: Grayscale Image

Fig. 3: Sharpened Image

D. Blurring

Blurring the image helps reduce the noise thus increasing the performance of plate localization. In this project, Gaussian filter, equation 3 was used to blur the image. Figure 4 show the output of image blurring.

$$G(x,y) = {\frac{1}{2\pi\sigma^2}}^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3}$$

E. Edge Detection

Canny edge detection was used to detect the edges in the images. This method proven to minimize the number of false positive and achieve optimal edge localization. Figure 5 show the result of Canny edge detection.

F. Skeletonize

Skeletonize is the method to thinning the edges aggressively until the contour of the edges reaches one pixel width. Figure 6 show the result of skeletonize.

G. Morphological Operations

Morphological operation strengthen the structural characteristics of the images. In this project, 10 combination of

Fig. 4: Blurred Image

Fig. 5: Edge Detection Image

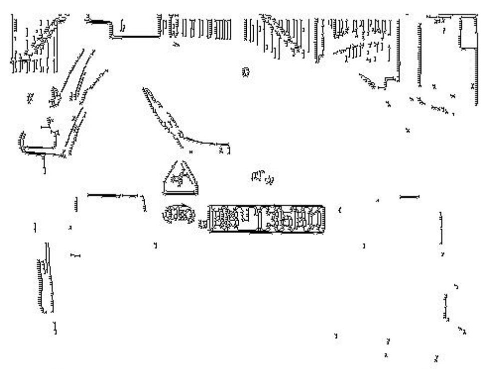

Fig. 6: Skeletonize Image

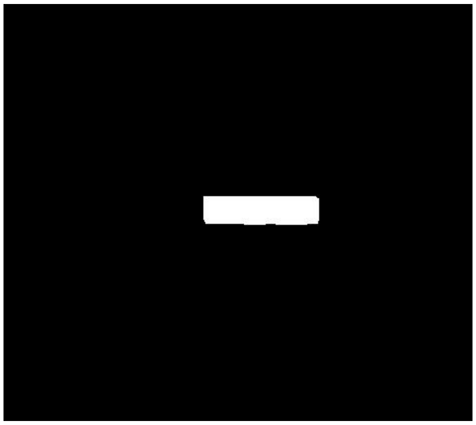

Fig. 7: Image after morphological operations

Fig. 8: Image after CCL

morphological operation was used. Figure 7 show the result of morphological operations.

1) Fill hole, opening, closing, fill hole x10, dilation x10
2) Closing, dilation x3, erosion x10, dilation x7
3) Closing, dilation x3, closing, erosion x10, dilation x7
4) Fill hole, dilation, erosion x10, dilation x7
5) Fill hole, closing, erosion x10, dilation x10
6) Fill hole, dilation, closing, fill hole, opening, erosion x10, dilation x9
7) Dilation, closing, erosion, opening, fill hole, erosion x10, dilation x10
8) Fill hole, dilation x3, closing, erosion x3, opening
9) Fill hole, opening, dilation x3, closing, fill hole, erosion x10, dilation x8
10) Fill hole, opening, erosion x5, dilation x5

H. Eight Neighbourhood Connected Component Labelling

Eight neighbourhood connected component labelling was used to identify the location of the rectangular shape, which may be a license plate in the image. There are 5 requirement for the images to be considered a license plate. The requirements are as follows:

$$R_{min} \leq \frac{width}{height} \leq R_{max} \tag{4}$$

where R_{min} is the minimum ratio and R_{max} is the maximum ratio. For this project, the ideal value for R_{min} is 2.9 while the ideal value for R_{max} is 6.

$$\frac{PC_{total}}{width \times height} \geq PCR \tag{5}$$

where PC_{total} is the total pixel count and PCR is pixel count ratio. For this project, the ideal value for PCR is 0.6.

$$PC_{min} \leq PC_{total} \leq PC_{max} \tag{6}$$

where PC_{min} is the minimum pixel count and PC_{max} is the maximum pixel count. For this project, the ideal value of PC_{min} is 20 and the ideal value for PC_{max} is 15000.

$$height \geq height_{min} \tag{7}$$

where $height_{min}$ is the minimum height. For this project, the ideal value of $height_{min}$ is 20.

$$width \geq width_{min} \tag{8}$$

where $width_{min}$ is the minimum width. For this project, the ideal value of $widthmin$ is 40.

V. Results

Our algorithm successfully localized 49 images out of 97.

Figure 9 show example of successful localization while figure 10 show example of failed localization.

Fig. 9: Example of successful localization

Fig. 10: Example of failed localization