Thai License Plate Recognition Using Proportional and Filtering Method

Kann Yingprayoon
kann.y@sse2015@tggs.kmutnb.ac.th
Department of Electrical and Software Systems Engineering
Thai - German Graduate School of Engineering
King Mongkut's University of Technology North Bangkok
THAILAND

Abstract

This paper is a preliminary study and development. We will focus on the detection of province name on the Thai license plate image and result the outcome in text. The procedure is to obtain a License Plate image and generate texts out of the lower resolution image. The image has to be cropped into the size of the plate which reveals only the image of texts. The image will be converted to grayscale first and then converted to binary after all. The process only needs binarized image. There are three main parts on Thai license plate which consist of alphabets, Arabic numbers and a province name. In order to detect a province name, we will consider just numbers and province name by collecting the size of the matrix and its values. Apart from detecting each letter, we will detect a whole word and save the values of its height, width, histogram, number of spaces between each letter, number of letters and number of vowels to create a proportion between each value and compare, evaluate and result the outcome into a certain province text. All the values will be stored in the program. Each province will be filtered by each proportional value to distinguish into different provinces. All the processes will be simulated using MATLAB software.

1. Introduction

According to a huge number of vehicles that have been increasing every single month. New cars try to present on roads whilst old cars still exist. This means that vehicle identification has to play a very important role in order to control traffic flow. In Thailand, traffic jam is everywhere and has caused a serious problem to the transportation. A license plate recognition can be one of the solutions. The cost that had been spent on the instruments for detection was very high. There is a potential to reduce the costs for the instruments by creating more efficient software to reduce costs and time. The procedure is to obtain a License Plate image and generate texts out of the lower resolution image. There are three main parts on Thai license plate which consists of alphabets, Arabic numbers and a province. Even though two license plates have the same numbers and alphabets, they can have different provinces. In this paper, we will focus on the province name in which there are 77 provinces and 2 non-provinces that are found on Thai license plate. The size of province name can be a problem for image recognition because of the size province name is much smaller than the size of alphabet or number (approximately 1:3 to the size of alphabet or number). This paper is a preliminary study, development and add-on from previous researches.

There are steps that we tried to experiment in this research:

- Image preparation
- Locate the numbers and the province
- Measure the height of numbers as pixels
- Measure the height of province and compare

2. Methods

These are the processes for image preparation

- Image cut and resize.
- Convert the image from RGB image to Grayscale image.
- Convert the grayscale image into a binary image.

2.1 Image cut and resize

In this research, we used the license plate that had been taken from our camera and manually cut to prepare the image for training data set

Figure 2.1.1 Thai License Plate

Figure 2.1.2 Thai License Plate components

The image will also be resized to the same 115 pixels of width in order to simulate low resolution situation.

Figure 2.1.3 Low resolution Thai License Plate I   **Figure 2.1.4** Low resolution Thai License Plate II

2.2 Convert the image from RGB Image to Grayscale Image

Figure 2.2.1 RGB image   **Figure 2.2.2** Converted from RGB to Grayscale

Convert the image from an RGB image with 3 channels of color into a grayscale image with a single channel color that varies from 0-255 by forming a sum from R, G and B components (see [2]):

$$0.2989 * R + 0.5870 * G + 0.1140 * B$$

2.3 Convert the grayscale image into a binary image

Convert the grayscale image into a binary image by creating a threshold and obtaining values from grayscale image.

The threshold that we use is constantly 50% of ceil(255) for the image with the width size more than 500 pixels and will keep increasing as the image's width is lower than 500 pixels with the rate of 10% every 400 pixels of image's width

```
if (Image's width) <= 500
    threshold = ((500-(Image's width)) * (10/400)) + 50;
else
    threshold = 50;
end
```
Grayscale image values vary from 0 to 255.

For example, the width of an image is more than 500 pixels, the threshold of the image will be 50% of ceil(255)

$$255 / 100 * 50$$
$$= \text{ceil}(127.5)$$
$$= 128$$

Then we can set values to the image, if the value of the image is more than 128, assign as 1 and if the value of the image is less than 128, assign as 0 in which 0 means black and 1 means white.

Figure 2.3.1 Grayscale image   **Figure 2.3.2** Binarized image

2.4 Find the height of "Numbers" and "Province" in pixels (including upper and lower letter)

In Thai language, there will be upper and lower vowel. Therefore, some of the province texts are hidden under the license plate frame. As shown in Figure 2.4.1

Figure 2.4.1 Letters hidden under license plate frame

As in Figure 2.4.1, the letter should be บุรีรัมย์ instead of บรีรัมย์.

Therefore, the image that we want in order to process and receive a result will be only half of the province's height.

Figure 2.4.2 Wanted part

Figure 2.4.3 Wanted part II

In order to know the size of Numbers (see Figure 2.1.2), we have to scan and look for the row where the first black pixel starts and measure the size.

Figure 2.4.4 Find the location of "Numbers"

The procedure to know the position of the space between Numbers and Province (see Figure 2.1.2) is to accumulate the value of pixels in each row as a white pixel has a value of 1 and a black pixel has a value of 0 and save those accumulated values in a column. In the column, we will look for the values that equal to the size of image's width. The location will be gathered as highest peaks from graph.

Figure 2.4.5 The histogram of accumulated values in each row

After we know the location where the Numbers (see Figure 2.1.2) is, then we can calculate the height of Numbers

Then we will find the height of Numbers (see Figure 2.1.2) in order to know the height of Province (see Figure 2.1.2) because as we have already measured the ratio between the height of "Numbers" and "Province" for every plate which is 1.7174:1
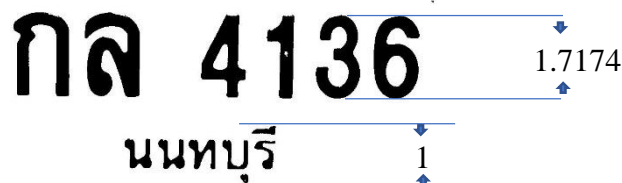

Figure 2.4.6 Size comparison

But as we mentioned above, we will only use half of the province size. Therefore, the height of "Numbers" and half "Province" (see Figure 2.1.2) has a relationship in ratio of 3.4348:1
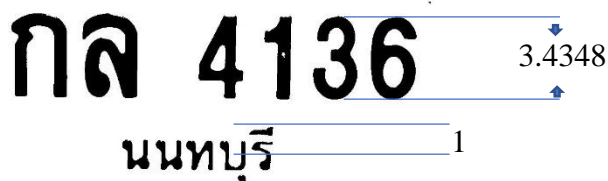

Figure 2.4.7 Size comparison between Numbers and half Province size

After we know the size of Numbers (see Figure 2.1.2), we will know the size of half Province where we can get data from.

Figure 2.4.8 The image of half province

2.5 Find the segmentation of half Province (see Figure 2.4.8)

The step of segmentation is to distinguish and count white spaces from black pixel letters by using graph to show the histogram of half Province in each column. The column that has the same value as the width of half Province (see Figure 2.4.8) image will be considered as a space.
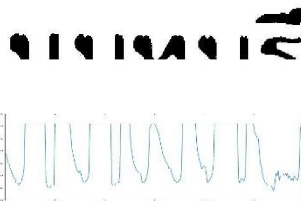

Figure 2.5.1 The histogram of half Province

The values will be counted as a number of peaks from graph.

2.6 Calculate and compare the values from half Province (see Figure 2.4.8)

There are 9 values that need to be calculated, compared for each province image.

1) The ratio between the number of black pixels and white pixels.
2) The ratio between Numbers's height size and Province's width size.
3) The result of 1) * 2).
4) The result of 1) + 2).
5) The ratio between the width of half "Province" image divided by the distance of the first highest peak from the left in x-axis.
6) The number of segmentation of half "Province" image.
7) The value of Numbers's height divided by the value of Province's width.
8) The number of segmentation of Upper Letter spaces.
9) The distance of Upper Letters in percentage to the width of Upper Letter image.

2.6.1 The ratio between the number of black pixels and white pixels

This step is to accumulate the value of black pixels and the value of white pixels. And then create a ratio between those values.

$$\text{Black pixels / White pixels = Result 1}$$

2.6.2 The ratio between Numbers's height size and Province's width size (see Figure 2.1.2)

Calculate the height of Numbers (vertical dimension) and the width of Province (horizontal dimension) from their location between space. And then create a ratio between those values.

$$\text{The height of Numbers / the width of Province = Result 2}$$

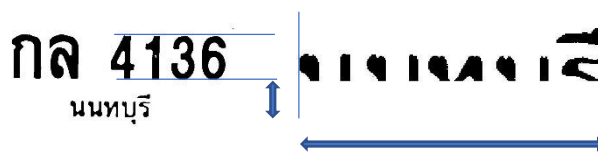***Figure 2.6.2.1*** *Number's height and Province's width*

2.6.3 The ratio between 2.6.1 multiplied by 2.6.2

Create an equation of 2.6.1 the ratio between the number of black pixels and the value of white pixels multiplied by 2.6.2 the ratio between Number's height size and Province's width size.

$$\text{(Black pixels / White pixels) / (The height of Numbers / the width of Province) = Result 3}$$

2.6.4 The ratio between 2.6.1 added by 2.6.2

Create an equation of 2.6.1 the ratio between the number of black pixels and the value of white pixels added by 2.6.2 the ratio between Number's height size and Province's width size.

$$\text{(Black pixels / White pixels) + (The height of Numbers / the width of Province) = Result 4}$$

2.6.5 The ratio between the width of half "Province" image divided by the distance of the first highest peak from the left in x-axis

Measure the width of half Province and find the distance of the first highest peak from graph.
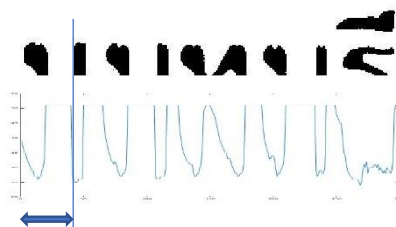

Figure 2.6.5.1 *Find the first highest peak*

$$\text{The width Half Province / the distance of the first highest peak = Result 5}$$

2.6.6 The segmentation of half "Province" image

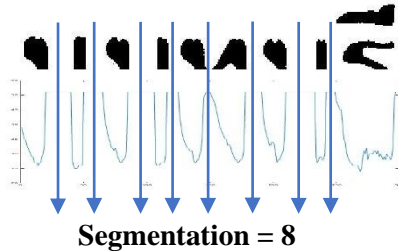Count the number of highest peaks and save the result as Result 6

Segmentation = 8

Figure 2.6.6.1 *Find the number of highest peaks*

2.6.7 The number of segmentation of Upper Letter spaces

Figure 2.6.7.1 *Upper Letters*

Figure 2.6.7.2 *Needed row*

The location of the row that we need is located as half Province's height / 6.4
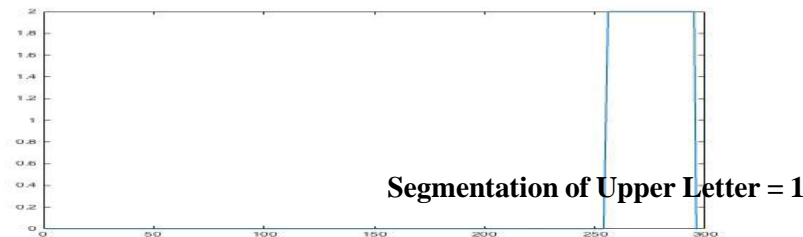

Segmentation of Upper Letter = 1

Figure 2.6.7.3 *Segmentation of Upper Letters*

(In Figure 2.6.7.3) The graph has been inverted to be easier to see. Then we will find the segmentation of the row (see the solution in **2.5**). The segmentation in this step is not finding spaces but to find letters. Save the result as Result 7

2.6.8 The distance of Upper Letters in percentage to the width of half Province

The width size of half Province image (see Figure 2.4.8) will be set as 100% while the percentage of Upper Letters depends on the distance. The most left one is set to 0% and the most right one is set to 100%. Save the result as Result 8.

2.6.9 Save all the result as a certain province

Gather all 79 provinces of Thai license plate example and get all the results for each province. Each province must have the same 8 results with 5% tolerance in each result. Therefore, in order to find out which province, the input image must meet the requirement of all 8 results which will filter and differentiate images into a certain province.

3. Experimental results

There are many parts in a license plate which can be categorized into three main parts 1) random letters 2) random numbers and 3) province/city name. In the case of random numbers and random letters, for instance, with English letter, the possibility for one letter in a certain position will equal to 1 in 26^1 which means that in one letter on a license plate, the letter could be any letter from A-Z. The possibility will be increased as the number of the letter's positions are found. For example, if two positions of the letter are found, the possibility will be 1 in $26^2 = 676$. Compare with the province/city name which comes in a word or an abbreviation. There is only one position for a word on a license plate. The possibility is most likely lower than those random letters/numbers which makes province/city name easier to predict. Many other methods tend to scan the plate letter by letter. Therefore, this method in this paper has been created to store all the data from the license plates that has been tested in our method in order to predict province/city name.

With almost 79 tested provinces, the accuracy in the research depends on training as the result of each province may varies on the quality, the brightness, the resolution of the image and even also the obstacles such as dirt or the reflection of lights. This method proved in effective time consumption for each image recognition as an average as 750ms per image.

3.1 The comparison of the letter by letter scan method
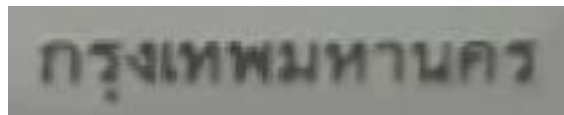The process is tested in ABBYY FineReader program

Figure 3.1.1 *Example of tested Thai license plate*

The result from FineReader is กรุงเทพมพานคว while the correct one is กรุงเทพมหานคร. Some of the letters are misread, showing that the letter by letter scan method is not efficient enough.

4. Conclusion

New approach of Thai license plate recognition has discovered their efficiency in both time consumption and accuracy in this paper. This preliminary research is still in development and can be improved further to be more effective and efficient.

References

[1] Sumanta Subhadhira, Usarat Juithonglang, Paweena Sakulkoo. (2014). *License Plate Recognition Application using Extreme Learning Machines*. Summitted to the 2014 Third ICT International Student Project Conference (ICT-ISPC2014).

[2] Vidhya Suhan L Nathan, Ramkumar J, Kamakshi Priya S. (2004). *New approach for license plate recognition system*. Submitted to ICISIP.

[3] Yasuo OGIUCHI*, Masakatsu HIGASHIKUBO, Sakda PANWAI, Ekarin LUENAGVILAI. (2014). *Automatic License Plate Detection and Recognition in Thailand*.

[4] P. Sa-ngamuang, C. Thamnittasana, T. Kondo. (2007). *Thai Car License Plate Recognition Using Essential-Elements-Based Method*. Proceedings of Asia-Pacific Conference on Communicationa 2007.

[5] Sumanta Subhadhira, Usarat Juithonglang, Paweena Sakulkoo. (2014). *License Pale Recognition Application using Extreme Learning Machines*. Submitted to The 2014 Third ICT International Student Project Conference (ICT-ISPC2014).

[6] https://www.abbyy.com/en-apac/finereader/