Digital Image Processing - Final Project Plate recognition

Maria Alejandra Aguiar 1455775 and Luis Felipe Castano 1455721

Abstract—Over years, technology has been used to develop applications that aim to simplify the human task, to execute tasks efficiently, faster and with quality. One application is the license plate detection system. License plate detection can be used for toll processing, at parking areas and for traffic monitoring purpose. The biggest challenge is that image processing of license plates needs to be quick and accurate. In this project, we use a three stages process: (i) Applying preprocessing to license plates images to separate the plate number from the background and noise, (ii) Separating characters from the license number, (iii) Using each character as a input to the SVM for classification. Experimental evaluation, using 66 images extracted from A.L.P.C.A dataset, indicates that the proposed approach correctly classified 80,77% of the characters from the license plates.

Index Terms—Digital Image Processing, Automatic Number-Plate Recognition, Classification, Suport Vector Machine

1 Introduction

Digital image processing is an important field in computer science, since this field has several practical applications that improve quality of life and solve many problems of daily life. This is the case of Automatic number-plate recognition (ANPR), a subject studied for more than 50 years in the field of computer vision, the applications of ANPR are: security, collection of electronic tolls, traffic monitoring, registration in parking areas, among others. This document contains: related work, an explanation of the proposed method as final project of digital image processing course and finally shows the results and conclusion obtained.

2 Related work

Since we saw the concept in class and found it the most suitable to use in this project. We review the use of support vector machine concept in license plate recognition. Support vector machine (SVM) is a supervised learning method used for classification and regression. In simple words, given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that predicts whether a new example falls into one category or the other.

2.1 SVM Based License Plate Recognition System [2]

A number plate recognition algorithm is proposed for character segmentation and recognition. This algorithm employs an suport vector machine to recognize numbers. Each character is recognized by an SVM, which is trained by a collection of samples of numbers from number plates. In order to recognize a number plate correctly, all numbers are tested one by one using the trained model. The recognition results are achieved by finding the maximum value between the outputs of SVMs.

2.2 Automatic Car License Plate Recognition System using Multiclass SVM and OCR [3]

In this paper, the authors implement a system of car license plate detection and recognition system using SVM and OCR. They propose the usage of a multiclass SVM for the feature extraction of the license plate. Integrated with the optical character recognition.

2.3 Automatic recognition of a car license plate using color image processing [4]

An automatic recognition method of a car license plate using color image processing is presented in this paper. At first, background colors of a plate are extracted from an input car image. A neural network is used for more stable extraction. To find a plate region, a fixed ratio of horizontal and vertical length of a plate is used. To recognize characters in a plate, template matching and postprocessing techniques are used.

3 Description of Algorithm

Two basic stages are defined for the implementation of this algorithm: (i) preprocessing of an image, where the noise is eliminated, correct shadows is applied and characters are segmented; (ii) Previously selected characters are classified by a Support vector machine. Process is described in detail below:

3.1 Prepocessing

For preprocessing is defined six stages:

3.1.1 First stage: Contrast enhancement

This operation improve an image contrast to obtain an uniform histogram. In this case a contrast limited adaptive histogram equalization (CLAHE) is applied, obtaining the following result:

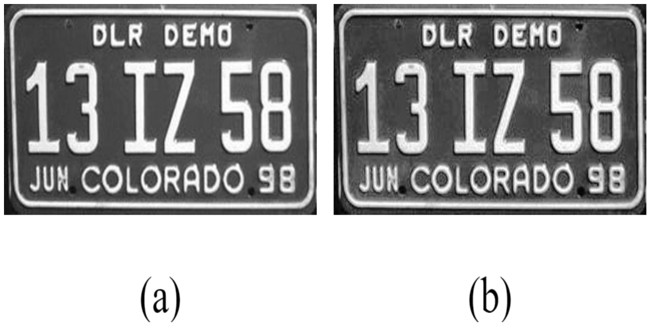


Fig. 1. (a) Green channel of the original image (b) after CLAHE.

3.1.2 Second stage: Shadow correction

The shade-correction algorithm is observed to reduce background intensity variations and enhance contrast in relation to the previous image. For this a background image is created using the following process: (i) a mean filter with a 3x3 kernel, (ii) a Gaussian filter with a 9x9 kernel, (iii) a media filter with a 40x40 kernel. The resulting image is subtracted from theimage obtained previous in order to corrected shadows.

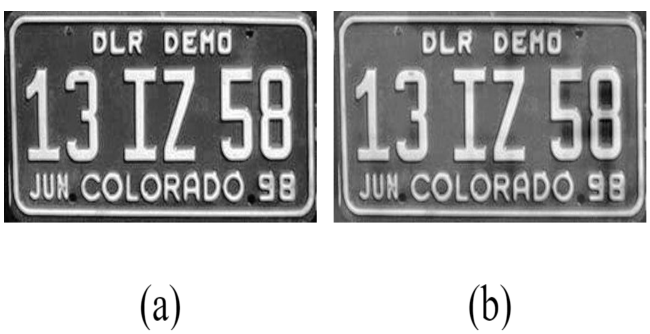


Fig. 2. (a) CLAHE image (b) Shadow corrected image.

3.1.3 Third stage: White objects conversion

Since some plates have many color variations, interest objects (characters) could suffer an inversion of colors. This is a limitation when processing plates. For that reason, histogram of an image is used to identify background, if background is shifted to the left (low values) the image is inverted.

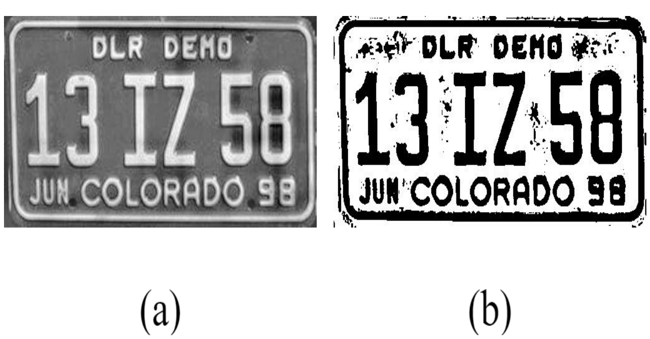


Fig. 3. (a) Shadow corrected image (b) White objects conversion.

3.1.4 Fourth stage: Simple noise elimination

All the images of processed plates are transformed, converting an an image of any size into an image of a size of 500 x 250 pixels. Taking into account above, it is possible to apply a simple noise elimination by identification of the size of the objects, any object that does not have a minimum size K is eliminated.

Fig. 4. (a) White objects conversion (b) Simple noise elimination.

3.1.5 Fifth stage: Elimination by radius

Some objects persist after simple noise elimination since they have similar characteristics to objects of interest (they occupy similar amounts of pixels in an image). To eliminate persistent objects, we calculate the average radius of the minimum enclosing circle of all the objects present in the image, then median of these values is taken as a reference point and a standard deviation of 10 pixels is established (this standard deviation is calculated with maximum and minimum value of radius of the characters present in the test images). Finally, all the elements that do not have a radius equal to the median of the radius of the current image with a standard deviation of 10 pixels may be eliminated.

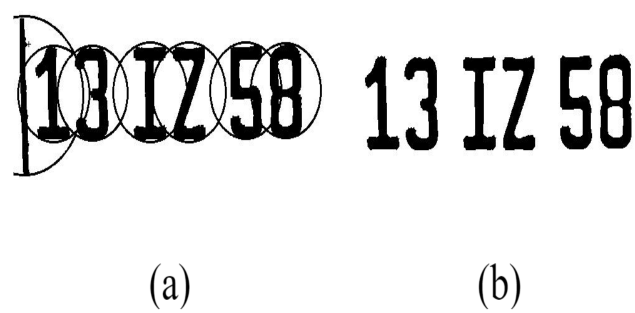


Fig. 5. (a) Simple noise elimination with radius (b) Elimination by radius.

3.1.6 Sixth stage: Character segmentation

Finally, segmentation of characters is achieved by applying boundary rectangle . See Fig 6

3.2 SVM Classifier

In this project, the SVM classifier of SCIKIT-LEARN (Machine Learning Python library) was used due to its efficiency and simplicity for data analysis. Since the purpose for the SVM, in this case, is to classify license plates character by

Fig. 6. (a) Elimination by radius (b) Character segmentation.

character, a dataset called Chars74K [5] containing about 62.000 images of characters from computer fonts with 4 variations (combinations of italic, bold and normal) was selected for the training. Then, 66 images from a public license plate dataset called A.L.P.C.A [6] were selected for validation.

Before training the SVM, it is necessary to perform a serie of operations on the images. First they are vectorized and then, converted to boolean. This as a requirement to be able to pass them as an input to the SVM and to improve results and efficiency. When the training is finished, the model is saved and it is subsequently loaded for validation.

4 Experimental Results

The proposed method is implemented using Python. We perform our experiments on segmented single characters of 66 license plates images previously selected. Some resulting images with their respective prediction from the SVM are shown in Table 1.

Finally, we obtain an accuracy of 80,77% shown in Fig 7, from the validation images. With the different results, we noticed that despite the great amount and variety of the training set, characters such as Q, O, D and 0 are confused with each other by similarity in shape, as well as 6, B and 8 or 2 and Z. For which precisely accuracy decreases.

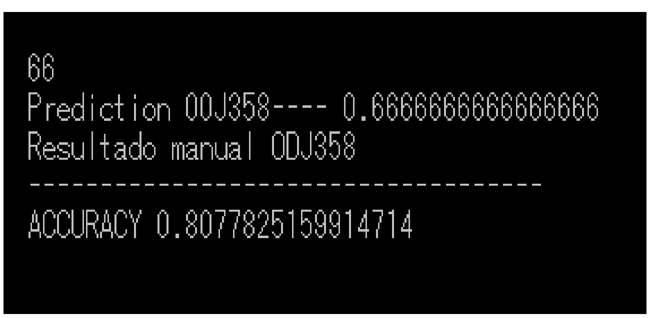
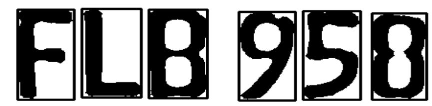
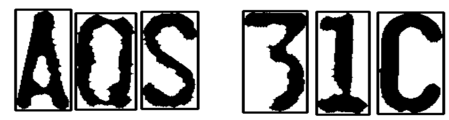
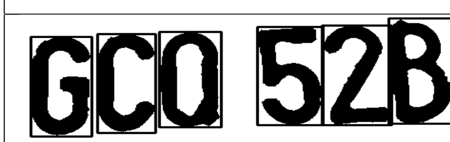
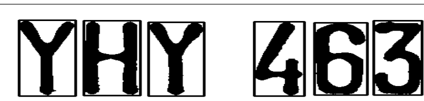
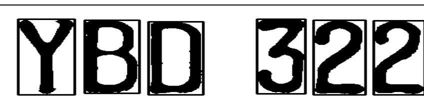


Fig. 7. VSM accuracy with the validation set

TABLE 1
Results

Result Images	N Predicted
	CJP118
	FLB950
	TTK4Z8
	A0S31C
	OYZ03Z
	646AVZ
	6CO5Z6
	YHY463
	DZA149
	YBO3ZZ

5 Conclusion

The method proposed in this paper is suitable for license plates of different countries and despite that some images have low contrast or illuminated areas, it achieves acceptable results.

To obtain high accuracy in classification of characters, it is necessary a dataset with a large number of images for training. In this project, the classifier was trained with a set of 36.500 characters, obtaining an accuracy of 0.80

The characters with similar shapes are especially difficult to differentiate for the classifier. Our classifier had difficulties to differentiate some characters, for this reason the performance may be improved if a transformation of the frequency domain is used in the characters before classification

The precision of classification of characters depends completely on the quality of the preprocessing applied to an image, if one stage of the preprocessing is not suitable for the problem addressed, it could lead to irreversible failures in classification.

The elimination of objects based on its minimum confinement radius proved to be a good method of noise elimination, thus obtaining segmentation only of the objects of interest in the 66 validation images processed in this project.

6 BIBLIOGRAPHY

REFERENCES

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.
[2] K. Parasuraman and P. Subin, SVM Based License Plate Recognition System, Itfrindia.Org, 2010.
[3] R. Madhukar, Automatic Car license plate Recognition system using Multiclass SVM and OCR, vol. 30, no. 7, pp. 369373, 2015.
[4] Eun Ryung Lee, Pyeoung Kee Kim and Hang Joon Kim, "Automatic recognition of a car license plate using color image processing," Proceedings of 1st International Conference on Image Processing, Austin, TX, 1994, pp. 301-305 vol.2.
[5] Ee.surrey.ac.uk. (2018). The Chars74K image dataset - Character Recognition in Natural Images. [online] Available at: http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/.
[6] Alpca8123.50webs.com. (2018). The License Plate Gallery. [online] Available at: http://alpca8123.50webs.com/.