OpenCV 3 and Support Vector Machine method in car-plate recognition application

Maribel Jaramillo, Juan Osorio, Natalia Rubiano
Computing engineer students.

Abstract—OpenCV functions allows to locate the numbers and letters of a car plate, after the identify them, will be read to recognize them using a Support Vector Machine (SVM), this method, is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes the input data, in this way, the numbers and letters of the car plate will be recognized.

Keywords—OpenCV, Support Vector Machine, Car-Plate recognition, Python.

1 Introduction

License plate recognition has a huge range of applications, which use the extracted plate number to create automated solutions for a lot of problems. These include the following topics: Access control, tolling, border control, Traffic control, to find out the stolen cars. The objetive is to locate the number plate using a morphological operator (Rectangle). The located plate is preprocessed to remove the noise and the result is passed to the segmentation part to segment the individual characters in the license plate after segmentation the output is normalized and passed to OCR algorithm. Finally the characters are recognized using template matching algorithm, in this case, Support Vector Machine.

2 Prior and related work

Optical character recognition (OCR) is a technology that automatically identifies the text contained in an image, then transforms it into a string of alphanumeric characters to interact with them in text editing programs.[1] In OCR systems, at least 4 stages are distinguished: [2]

- **Preprocess:** Image adequation.
- **Segmentation:** Selection of the zone of interest.
- **Features extraction:** Digital representation of the image.
- **Recognition:** Distinction of the character contained in the image.

There is a wide variety of applications that make use of OCR to:

- Input of data for business documents, such as invoices, purchase orders, receipts.
- Verification of document such as passport.
- Automatic recognition of vehicle registration plates.
- Convert handwriting in real time, such as the digital pen that functions as the input interface.
- Digitization of books.

In the case of an automatic recognition of vehicle registration there are several jobs that have been implemented by this technology, some of these are:

Automatic license plate recognition: [3]

In this work, the characteristics of the system have the same relationship, as the reflectance, the intensity and the variation of the values of each RGB color pixel. They perform a vertical location, using the components of the saturation and, later, a horizontal location in which it makes use of the intensity, the latter does not apply to the images with a flat value in the intensity. Followed by this and taking into account the cutting of the plate, performed

the process of binarization and segmentation, separating the bottom of the elements and cleaning the impurities in the image. Then a process of rearrangement of letters taking into account that there are plates of two lines. Finally, the process of character recognition is performed, making use of perceptrons, in which their training is based on supervised learning.

Prototype of a vehicle image acquisition system, automatic detection and recognition of the characters of the plate in real time through artificial vision, applying vehicular control: [4]

In this project a study was carried out from the acquisition of the image, the processing of the image until the detection and extraction of the characters of the plate. For the detection of the plate they implemented several algorithms: segmentation by thresholding; location through projections; and locating the area with the highest reflectivity. For character recognition, a preprocessing is first performed to eliminate possible areas that are in the image and do not belong to the information that you want to know; then the segmentation process is done and finally the recognition of the characters. Taking into account that the work is focused on plates of Ecuador, the characters are separated according to the structure of these, the first 3 characters are letters and the remaining 3 or 4 characters are numbers. For the recognition of the characters is made through the correlation process, comparing each character of the plate with character templates stored in the database.

3 Algorithm Description

Considering that it is not necessary to locate and remove the plate from the image, because the application receives the image of the plate directly for its prediction, the steps for the recognition of plate numbers are:

3.1 Read the image and resize it to 500 * 300 pixels.

Fig. 1: Resized image.

3.2 Convert the image to grayscale.

Fig. 2: Gray scale image.

3.3 Eliminate noise: Through the media filter, the possible noise of the image is reduced.

Fig. 3: Image with media filter.

3.4 Trim the background: Taking into account that the plate can have a small background, it is done:

- Thresholding, through the Otsu method, which allows a binary image to be obtained.

Fig. 4: Image with Otsu method.

- The drawing of rectangular external contours of the image, because the plate has a rectangular shape.

Fig. 5: Image with external contours drawn.

- The clipping of the image according to the dimensions of the largest area of the contours found. It should be noted that the largest area will always belong to the outline of the plate.

Fig. 6: Cropped image.

3.5 Character segmentation

- The opening filter is applied with the purpose of smoothing without altering the size.

Fig. 7: Image with opening filter.

- The dilation filter is applied, in order to highlight the characters of the image.

Fig. 8: Image with dilation filter.

- The colors of the image are inverted, with the purpose of facilitating the search of contours.

Fig. 9: Image with colors inverted.

- Next, the contours are searched and those that comply with height, perimeter and area conditions are drawn.

Fig. 10: Image with contours drawn.

- Finally, each contoured element is stored as a 20 * 20 image in a list to be able to predict its equivalence.

Fig. 11: Segmented image.

3.6 Training

For the identification of the plate number, the sklearn Python library is used, this library facilitates the use of the SVM classification algorithm. Also, the way in which they do the training in [5] was studied, which was very helpful to be able to perform the prediction of characters.

The file SVM.py is responsible for the training, first save all the characters that the plate can have, then read all the images of each class that represents a character, store this data in two arrays and then create a linear type model, in which the probabilities were established in true assuming that the prediction is correct.

Now, most of the functions used are those that the library has by default.

Finally, the execution of this file saves the model created with a .pkl extension

3.7 Prediction

First the file of the model is read, then, with the predetermined function predict() of the sklearn library, each image is called and the character is predicted, thus obtaining the value of the plate.

However, the identified characters are kept in disarray. To order them use the value of the position x of the rectangular contour of each character, these values are ordered from lowest to highest, thus, it facilitates that the identified characters can be sorted.

4 Experimental results

4.1 First test - Input:

Fig. 12: Input image.

- Output:

Fig. 13: Image with opening filter.

- Console output:

```
Numero de Placa: CZR263
```

Fig. 14: Image with opening filter.

4.2 Second test - Input:

Fig. 15: Input image.

- Output:

Fig. 16: Image with opening filter.

- Console output:

```
Numero de Placa: CFD682
```

Fig. 17: Image with opening filter.

4.3 Third test:

Fig. 18: Input image.

- Output:

Fig. 19: Image with opening filter.

- Console output:

```
Numero de Placa: MSP253
```

Fig. 20: Image with opening filter.

5 Conclusions

1) The accuracy of recognition is the most important in this system. Therefore, the application must be optimized and modified to overcome some limitations to make the recognition more accurate, some preprocesses must be added to eliminate interference.
2) For a good recognition of plate characters, a good segmentation of the image is necessary, since there are several types of plates with different colors, noise, contrasts and inclinations, it is more difficult

to carry out the segmentation process, that is why it can exist cases that the characters on the plate can not be detected.

3) It was observed that there are many characters with similarities, such as the letter O with the number 0, the B with the 8, which makes it probable that sometimes the prediction is not so accurate, if the algorithm was used for Colombian plates a solution to improve the result would be to take into account the order in which the series of the plate is composed, that is, the plate initially has 3 letters and then 3 numbers, in this way the letters would be compared with the information of letters and numbers with the stored information of numbers.

REFERENCES

[1] Pablo de Olavide University. *¿Qué es el OCR (Reconocimiento óptico de caracteres)?*. [Online]. Available: https://www.upo.es/biblioteca/servicios/inst_equip/lab/materialesapoyo/manuales_hardware/escaner_ps5000c/pagina_16.htm

[2] C. Sánchez and V. Consuegra. *Reconocimiento Óptico de Caracteres (OCR)*. [Online]. Available: http://www.it.uc3m.es/jvillena/irc/practicas/08-09/09.pdf

[3] C. Parra and D. Regajo. (2006). *Reconocimiento automático de matriculas*. [Online]. Available: http://www.it.uc3m.es/jvillena/irc/practicas/06-07/14.pdf

[4] C. Bentancourt, P. Haro and G. Marcelo. (October 2011). *Prototipo de un sistema de adquisición de imágenes de vehículos, detección y reconocimiento automático de los caracteres de la placa en tiempo real por medio de visión artificial, aplicado al control vehicular*. [Online]. Available: http://www.it.uc3m.es/jvillena/irc/practicas/06-07/14.pdf

[5] F. Oladeji. (August 2017). *Developing a License Plate Recognition System with Machine Learning in Python*. [Online]. Available: https://blog.devcenter.co/