OCR Service Platform Based on OpenCV

Jianjun Cai¹, Erxin Sun², Zongjuan Chen³
China Automotive Technology Research Center Co., Ltd.
¹ caijianjun@catarc.ac.cn
² sunerxin@catarc.ac.cn
³ chenzongjuan@catarc.ac.cn

Abstract. This paper built an OCR service platform based on OpenCV. Cut images of license plates and characters were generated by analyzing and processing the images based on the OpenCV algorithm and utilizing the image big data accumulated in the production of application software in the automobile industry. Label these cut images as training data and input SVM, ANN and other trainers to train the model to recognize the characters in the new license plate image. In addition to license plate recognition, VIN code recognition, driver's license number recognition and other OCR functions can also be developed in a similar way. The platform is developed based on Java language and SpringBoot framework in order to be technically unified with other automotive industry application software. It needs to use Java to call OpenCV c++ interface implementation, and the platform provides rest interface externally.

1. INTRODUCTION
In the automotive industry application software, the system often needs to read information such as license plate number, VIN code, driving license number, etc. The input source of these information is often pictures. If the text information in the picture is read manually and then entered into the system, the workload is large and the error rate is high, and the automation of the business process cannot be realized. Therefore, the business demand for OCR services is increasing. However, due to many technologies and computer languages that could implement OCR, if each system develops its own OCR function, the cost is very high; calling the API interface of a third-party company requires more money and usually cannot meet individual needs and extensions.

The automotive industry application software has accumulated a large amount of industry big data including images in the long-term production. Use these data to extract more advanced value has become a voice. Business hopes to establish a unified service platform, use automobile big data to train the required algorithm models, and provide various intelligent service interfaces externally.

This paper designed an OCR service platform for software companies in the automotive industry. The platform can analyze and process existing image data by using SVM, ANN and other algorithms for training, and generate models to meet license plate recognition, VIN code recognition and various new functions in the future on demand. Various intelligent services to external systems were provided through the Rest interface. This article mainly focused on the functional algorithm of license plate recognition.
2. BASIC CONCEPT

OCR (Optical Character Recognition) refers to the process in which electronic equipment analyzes and processes image files, recognizes the characters in them, and converts them into text format.

When the image seen by the human eye is converted to a digital device, the value of each pixel of the image is stored, and the internal representation of the computer is a digital matrix. opencv is an open source cross-platform computer vision and machine learning software library, which contains the realization of some complete vision processing algorithms, as well as machine learning algorithms such as svm, with high recognition rate. The digital matrix of the operation image can be processed to obtain more advanced information.

Mat\(^4\) is the abbreviation of Matrix. The Mat class is an encapsulation class introduced by opencv for image processing, which basically covers the basic requirements of computer vision for image processing. The Mat class consists of two parts of data: a matrix header and a pointer to the matrix that stores all pixel data.

SVM(Support Vector Machine) is a pattern recognition algorithm, originally for binary classification. It is a supervised machine learning\(^5\) algorithm, which requires labeled data\(^6\) to train this algorithm, and each data set needs to have a category. If the feature vector of the instance is mapped to some points in the space, SVM creates one or more hyperplanes to distinguish each type of point. The system uses SVM model to train and determine whether the intercepted block is a license plate.

ANN\(^7\)(Artificial Neural Network) refers to a complex network structure formed by a large number of neurons connected to each other. It is an algorithm model based on simulating the structure and function of the biological nervous system. MLP (Multi-Layer Perceptron) is the most commonly used type of ANN. MLP consists of an input layer, an output layer and one or more intermediate hidden layers. The input layer is passed in by the feature vector of the training sample. Each layer contains several neurons, and the neurons are connected by arcs with variable weights. ANN changes the weight and other parameters through algorithms such as gradient descent and backpropagation through learning the input training data to achieve correct prediction of new data. This system uses the ANN model to train and recognize license plate characters.

3. PLATE RECOGNITION

Plate recognition is divided into two processes: plate detection and character recognition:

(1) Plate Detection

The plate detection process is to detect the position of the license plate in the image, including three sub-processes: plate location, SVM training, and plate judgment. Because the amount of original license plate image data is very large, it will be very slow to directly perform character recognition on it, so it is necessary to try to reduce the amount of calculation in the license plate recognition process. A series of different filters, morphological operations, contour algorithms and verification algorithms are used to preprocess the image containing the license plate to extract the image block that may contain the license plate.

Figure 1. GaussianBlur
The steps for image preprocessing and segmentation that may contain the license plate are as follows:

1) Gaussian blur
   This step mainly denoises the image to prevent the noise from being detected as an edge. After this step, it can be seen that the image has become blurred.
   The corresponding function in opencv is GaussianBlur().

2) Grayscale processing
   This step prepares the grayscale environment for the edge detection algorithm, and all subsequent operations are not based on color images. Because of the large amount of color image information, many image processing algorithms, such as sobel, are only suitable for grayscale images. To facilitate calculations, we need to convert color images into grayscale images. On the basis of preserving the image outline and features, the grayscale image can still reflect the outline and texture of the entire image.
   The corresponding function in opencv is cvtColor().

3) Sobel operator
   This step is to detect the vertical edges in the image. After this step, the license plate is clearly distinguished. Assuming that there is a line in the image, the left side is very bright and the right side is very dark, the human eye can easily recognize this line as an edge. This is also the place where the gray value of the pixel changes rapidly, and it is reflected in the mathematical function, which is the place where the derivative is the largest. The principle of the sobel operator is to find the first-order horizontal and vertical derivatives of the image, and determine whether it is an edge according to the magnitude of the derivative value.
Because the image is two-dimensional, do horizontal and vertical sobel convolution on the original image to get the brightness transformation of each pixel in the horizontal and vertical directions. Finally, the two matrices are superimposed, and the intensity of pixel gray value changes in the horizontal and vertical directions are comprehensively considered to obtain the edge.

The corresponding function in opencv is Sobel(),addWeighted().

4) Binarization
In a grayscale image, the value of each pixel is a number between 0-255, which represents the degree of darkness. Binarization is to set a threshold to convert the grayscale image into two possible values of 1 and 0, which will make the entire image present a clear black and white effect. In digital image processing, the binary image occupies a very important position. The binary image of the image greatly reduces the amount of data in the image, which can highlight the contour of the target.

The corresponding function in opencv is threshold().

Figure 4. Threshold

5) Close operation
After this step, the license plate area is connected into a rectangular area.
The corresponding function in opencv is getStructuringElement(),morphologyEx().

Figure 5. MorphologyEx

6) Take the outline
This step is to find all the contours in the figure.
The corresponding function in opencv is findContours(),drawContours().
7) Filter outline

If there are multiple extracted contours, find the minimum bounding rectangle for each contour, discard the rectangle if its size does not meet the condition, then take the skew angle of the rectangle, and discard it if the angle is greater than the threshold (e.g., plus or minus 30 degrees).

The corresponding function in OpenCV is `minAreaRect()`, `getRotationMatrix2D()`, `warpAffine()`, `getRectSubPix()`, `resize()`.

8) Uniform size

Before importing the rectangular tiles into the machine learning model, first rotate the tiles appropriately to make them level, and then adjust the tiles to a uniform size.

---

**Figure 6. FindContours**

**Figure 7. Splitted Plates**

After preprocessing and segmenting all possible license plate parts of the image, we use the SVM algorithm to determine which segmented parts are the real "license plate". Manually categorize these tiles, label them with license plate and non-license plate labels, as positive and negative samples input to the SVM trainer. When a certain number is gathered, we can select 70% of the tiles as training data and 30% as test data. The program reads the training data and traverses to extract features. The specific operation is to first grayscale the read image, then binarize the image, count the number of pixels whose grayscale values are greater than the threshold in the horizontal and vertical directions, and then normalize them as the final feature. Finally, configure the SVM trainer and perform statistical training on the previously configured sample features, and finally generate an xml file as the SVM license plate judgment model to determine whether the license plate is or not.

OpenCV defines the `CvSVM` class for the SVM algorithm, and initializes the use of training data, categories and parameters. The `train()` function in the `CvSVM` class trains the classifier, and the `predict()` function makes predictions.

In the actual license plate detection process, we hand over the blocks that may be license plates to the trained SVM license plate judgment model for judgment, and automatically select the blocks that are actually license plates.

(2) Character Recognition

The character recognition process is to obtain the characters on the detected license plate. It includes three sub-processes: ocr segmentation (character segmentation), ann training, and character recognition. First do ocr segmentation, call opencv's image processing algorithm to grayscale the license plate image, identify the color of license plate, and then binarize it according to the color of license plate. Because we need to get the outline of the character, and the algorithm of the outline looks for white pixels, we change the parameters of the binarization function to change the white value to black and the black value to white. Finally, we detect the contours of all white areas and find their
smallest bounding rectangle. For each detected contour, we check the size and filter out those areas that are too small or have an incorrect aspect ratio.

Figure 8. Threshold plate

Figure 9. Auxroi character

After segmentation, ANN is used to recognize the characters of these small tiles. For the massive character blocks obtained earlier, select a certain proportion as the training data, perform manual classification, type which letter, number or Chinese character it belongs to, and input it into the ANN MLP model for training, obtain the ANN character recognition model, and generate an xml file.

In the actual license plate recognition process, the obtained new character blocks are input into the trained ANN model, and the most likely character represented by each block is predicted through the model, and the characters become the license plate string after sorting.

4. SYSTEM ARCHITECTURE AND RECOGNITION RATE

The OCR service platform is developed based on OpenCV. At present, most of the software in the automotive industry is developed based on Java language. In order to unify the technology, the OCR service platform is developed by using java language, so we need to call opencv C++ interface through Java. Java calls C++ interface in two ways: 1) org.opencv Official package, need to introduce project build path; 2) org.bytedeco.javacpp Package, which is an open source library, provides an efficient way to access local C++ in Java, and introduces coordinate dependency in Maven POM.

The platform uses springboot framework and maven project management tool to develop, adopts front-end and back-end separation form, front-end uses layui framework. The external system provides OCR identification service in the form of rest interface. When the client uploads the image to be identified to the platform, it needs to bring the unique image identification. After the platform has recognized the characters, the recognition results will be stored in the database, and the recognition results will be returned to the client through JSON format. If the client wants to recognize or re-identify the same image next time, it can take the unique identification.

Firstly plate recognition should have pictures, extract plate blocks, extract character cuts, and then carry out label training, and then recognize characters in reverse. The recognition rate is related to extraction algorithm, training algorithm and training result. In order to improve the recognition success rate, we need to consider the diversity of training samples and debug and optimize the training algorithm. The result of training is corresponding to the method of sample extraction.

Figure 10. Architecture
5. CONCLUSION
A large amount of big data including images has been accumulated in the long-term production by the automotive industry application software, which make it possible for us to use these data to generate higher value. The OCR service platform developed based on OpenCV used OpenCV algorithms to analyze and process data images in the automotive industry and generate predictive models to achieve OCR functions. For technical unification, Java language and Springboot framework were used to develop the platform, and org.opencv or javacpp package were used to call OpenCV c++ interface in Java language. Rest interfaces were provided to the outside. Vin code recognition or other new OCR requirements can also be implemented in a similar manner by using algorithm training data to obtain training models to predict new data.

References
[1] Ramshankar Y, Deivanathan R. Development of Machine Vision System for Automatic Inspection of Vehicle Identification Number[J]. International Journal of Engineering and Manufacturing, 2018, 8(2):21-32.
[2] Garg S, Kumar K, Prabhakar N, et al. Optical Character Recognition using Artificial Intelligence[J]. International Journal of Computer Applications, 2018, 179(31):14-20.
[3] Virginia P. How Data Will Transform Industrial Processes: Crowdsensing, Crowdsourcing and Big Data as Pillars of Industry 4.0[J]. Future Internet, 2018, 10(3):24.
[4] Modrzyk N. OpenCV on the JavaVM[M]/ Java Image Processing Recipes. 2018.Merkel D. Docker: Lightweight Linux Containers for Consistent Development and Deployment[J]. Linux Journal, 2014(239):76-90.
[5] Singh P. Supervised Machine Learning[M]/ Learn PySpark. 2019.
[6] Yi W, Lu M, Liu Z. Multi-valued attribute and multi-labeled data decision tree algorithm[J]. Pattern Recognition & Artificial Intelligence, 2011, 2(2):67-74.
[7] Decebal, Constantin, Mocanu. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science.[J]. Nature Communications, 2018.