INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

IJARIIT

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 7, Issue 4 - V7I4-1139)
Available online at: https://www.ijariit.com

Automatic number plate recognition using contours and Convolution Neural Networks

Adithya M.
adithyam75@gmail.com
Atria Institute of Technology, Bengaluru, Karnataka

Sumitha B. S.
sumitha.bs@atria.edu
Atria Institute of Technology, Bengaluru, Karnataka

Rahul K.
kossigirahul487@gmail.com
Atria Institute of Technology, Bengaluru, Karnataka

Nitish Kumar P.
meganitish@gmail.com
Atria Institute of Technology, Bengaluru, Karnataka

Pramod G. N.
pramodgn99@gmail.com
Atria Institute of Technology, Bengaluru, Karnataka

ABSTRACT

Image processing technology is used in Automatic Number Plate Recognition (ANPR). Automatic Number Plate Recognition (ANPR) is useful for identifying stolen vehicles, smart parking systems, and the use of automobiles in unlawful operations. Character recognition is the first step of ANPR, followed by character segmentation and localization. The technique uses contours and morphological processes to locate the number plate initially. We execute character segmentation after localization. Convolution neural networks (CNN) are used by a segmented character to recognize things because they are known to be good at it. The trained CNN model has an 85.31% accuracy rate.

Keywords— *Contours, Morphological Operations, Character segmentation, Convolutional Neural Network, Character recognition, Image processing*

1. INTRODUCTION

With the increasing number of cars on the roads and highways, we are confronted with a slew of issues such as car smuggling, car use in terrorist acts, and illicit operations. Automatic number plate recognition (ANPR) is a system for reading number plates that uses character recognition on images. ANPR has the potential to be a very precise technology for detecting number plates The computer vision technology used to recognize identification number plates is known as ANPR. ANPR consists of three phases.

(a) License Plate Localization
(b) Character segmentation
(c) Character recognition

The number plate might appear anywhere in an image in ANPR, and localization is the most important aspect of the system. As a result, we're employing morphological procedures to locate the number plate. Character segmentation is the process of identifying and separating characters. Characters encode, validate, and finally display alphanumeric characters on screen after segmentation. CNN is used to recognize segmented characters. This CNN model has a high level of accuracy when it comes to recognizing characters on a license plate. In image detection and classification, a convolutional neural network is exceptionally accurate. Using a convolutional layer, CNN extracts the feature. It also uses completely connected layers to categorize the features into multiple classes. With 85.31% accuracy, this CNN model is a good choice.

2. RELATED WORK

Many strategies for developing the ANPR system were offered. Because these documents have numerous benefits and drawbacks. To recognize license plate characters, various methods are utilized. Here are a few of the different ANPR approaches. Here are a few of them:

An artificial neural network (ANN) was used to perform ANPR[3]. The proposed feature extraction approach was utilized to extract the number plate from the supplied image. ANN was used to recognize extracted characters in the final stage. Character recognition accuracy is 85 percent with this technology. Another way for ANPR is mathematical morphology, which is an image processing technique.

The mathematical morphology idea is used to extract the region from the input image. The number plate was segmented using digital image labelling, and the characters were recognized using template matching.

The paper [5] focuses on number plate recognition using a KNN and SVM classifier for number plate categorization. The simplest method of character recognition is to use feature extraction, which extracts the features of each character [6].

We use a Convolution Neural Network in our technique, which provides great accuracy for recognizing license plate characters. In picture categorization and recognition, a convolution neural network is particularly accurate. A convolution layer is used by CNN to extract the feature. It also uses fully connected layers to classify the features into multiple classes. This CNN model has a high level of accuracy.

3. PROPOSED SYSTEM

This section introduces the suggested system for number plate recognition. The goal of this paper is to create an accurate ANPR system. The developed method is separated into three stages: plate localization, character segmentation, and recognition of characters.

(a) Image Input
(b) Image Pre-Processing
(c) License plate localization
(d) Character segmentation
(e) CNN Recognition

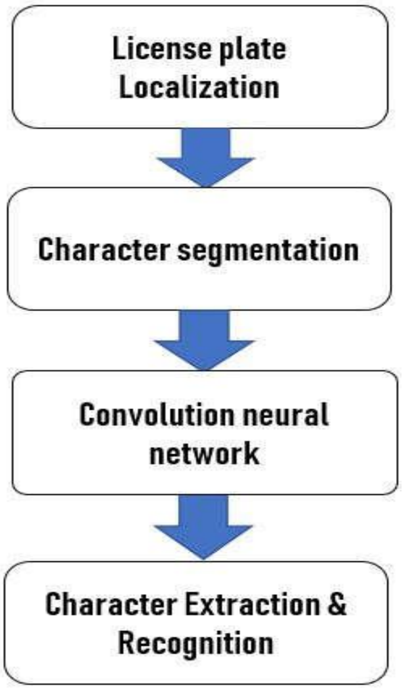

Fig 1. Proposed ANPR System

3.1 Number plate localization

Car Plate localization examines all of the pixels in a photograph to identify the registration code region. We start by converting the original image to greyscale and then enlarging it in pre-processing. To locate and detect license plates, we use morphological techniques. After resizing the image, morphological operators were used to enhance the number plate.

The opening and shutting morphological operators are used to remove noise from the registration number plate. Finally, find proper contours and authorize using geometrical conditions to extract registration code candidates. The following stages are involved in locating a license plate:

(a) Pre-Processing: Pre-processing is the process of turning an RGB colour image to a greyscale image in order to reduce complexity. Following the image's grey scaling, we resize it to reduce the number of pixels in order to achieve better and faster outcomes.

(b) License plate enhancement: For the number plate enhancement, we use morphological procedures, such as the top hat operator, which removes the shapes that don't fit with the structural element. The highest top hat operator suppresses the backdrop pixel to increase the supported shape and size of the number plate region. Following improvement, we use thresholding to separate the background and foreground pixels, allowing us to focus on the precise license number plate region.

(c) Noise removal and finding contour: Noise reduction and contour finding: For the localizing plate, we need to remove a large percentage of the image's backdrop. A filtering technique supported by morphological operations is intended within the system to eliminate these locations. Applying morphological opening and closing operations to smoothen and remove items from a picture in this scenario. These morphological processes shape the objects in the background and foreground while also removing minor objects.

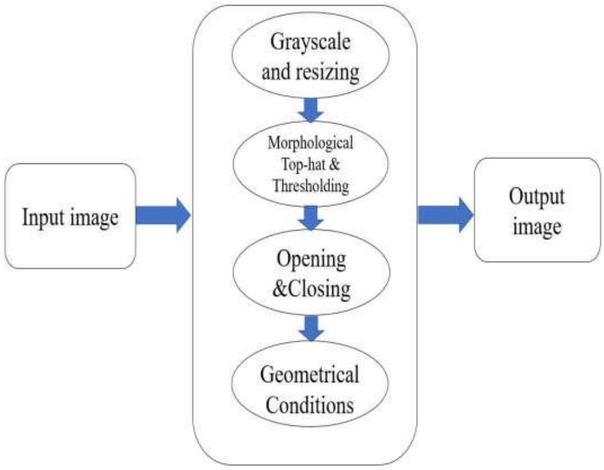

Fig 2. License plate Localisation Algorithm

Following noise removal, an image is composed of groupings of white objects. The contour detection procedure was used to detect the positioning of white items in a picture. The registration number plate's external boundaries are defined via contour detection.

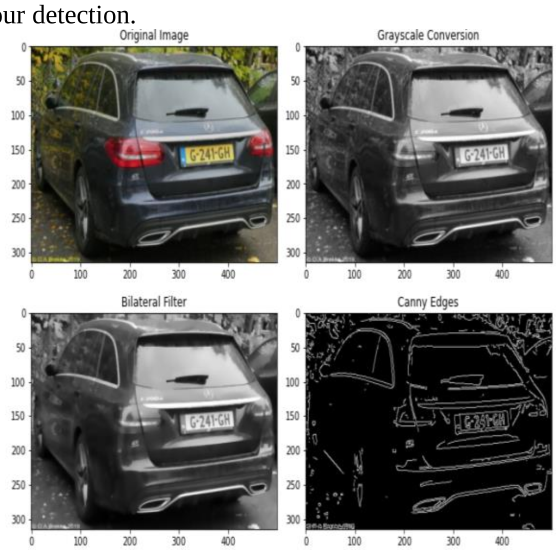

Fig 3. (a) original image (b) grayscale conversion (c) Bilateral Filter (d) Canny Edges

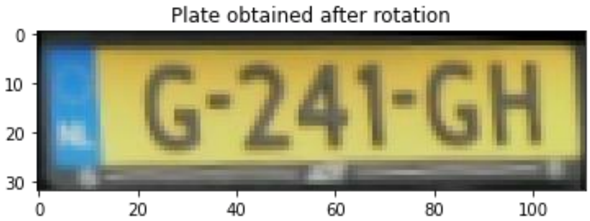

Fig 4. a) Plate obtained after rotation b) Extracted license plate

Fig 5. Detected license plate

3.2. Character Segmentation

Every alphanumeric character is separated during the character segmentation procedure. We segment characters for further processing after localization. The retrieved number plate is initially pre-processed in segmentation. We resized the number plate in pre-processing to produce a distinct and clear outcome of the number plate. We transform the RGB colour image to greyscale after scaling to reduce complexity. Using thresholding, a greyscale image is turned to a binary image, which can then be processed further. Boundary pixels and clean noise are removed from the final image using morphological procedures. Following morphological procedures, we define a list of dimensions with four values to compare the character's dimensions in order to filter out the required characters. The image is now clear enough to extract characters from the number plate.

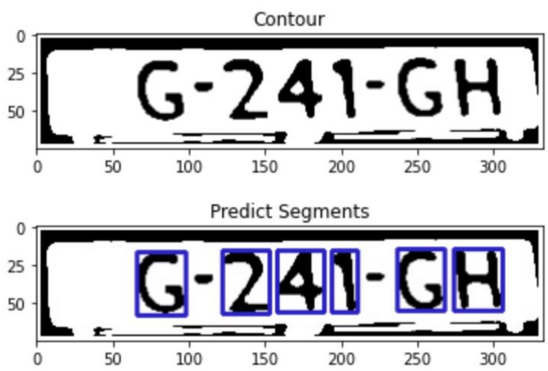

Fig 6. Contour and Predicted Segments

Fig 7. Individual Segments

6.3. Character recognition

The mathematical operation is performed by the convolution layer of CNN. Convolution is a highly specialized, linear operation. The convolution layer calculates matrix multiplication on CNN layers in convolutional neural networks.

(a) Input Layer: This is the layer from which we feed picture data into our model so that it can execute additional actions. The entire number of pixels in the data is divided by the number of neurons in this layer.

(b) Hidden layer: The generated data is passed as input data to the hidden layer. Many hidden layers exist, depending on our model and the amount of the input image data. The number of neurons in each hidden layer is more than the number of pixels. This hidden layer is made up of several layers. The output of each layer is calculated by multiplying the matrix of the preceding layer, and the network is then made nonlinear by employing the rely on activation function.

(c) Output Layer: The output from the hidden layer is fed into the output layer using logistic functions such as sigmoid and SoftMax. It turns the output into each class's likelihood score.

(d) Contours: The character contours derived from contour analysis are used to accomplish character matching.

(e) For each character obtained through character segmentation, the character recognition process is repeated. This procedure could be broken down into numerous stages. The output should be the number plate's recognised characters.

6.3.1 Contour Analysis: The contour analysis allows to describe, store, compare and find the characters presented in the form of the exterior outlines. It is supposed that the contour contains the sufficient information on the character shape. Interior points of the characters are not accepted in our system. The contour is the boundary of characters, a population of points (pixels), separating character from a background, which is obtained from thinning.

6.3.2 Thinning/Contour Extraction: The fundamental action is to make the boundary of characters 1 pixel thin. It observes the image pixel by pixel and erases the inner layers of black pixels on every character [6]. The image is observed repeatedly until every character boundary is reduced to single pixel thickness [7].

6.3.3 Convolution Neural Network: A convolutional neural network (CNN) is a form of artificial neural network that is specifically intended to process pixel input and is used in image recognition and processing. CNNs are image processing, artificial intelligence (AI) systems that employ deep learning to do both generative and descriptive tasks, often including machine vision, which includes image and video recognition, as well as recommender systems and natural language processing (NLP).

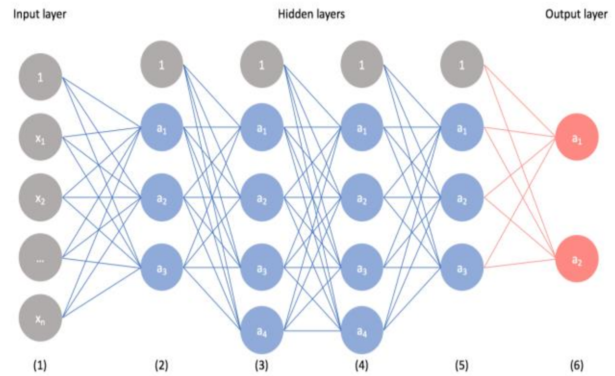

Fig 8. Convolutional Neural Network

A CNN employs a technology similar to a multilayer perceptron that is optimized for low processing requirements. An input layer, an output layer, and a hidden layer with several convolutional layers, pooling layers, fully connected layers, and normalizing layers make up a CNN's layers. The removal of constraints and improvements in image processing efficiency result in a system that is significantly more effective and easier to train for image processing and natural language processing.

Since the data is all clean and ready, now it's time to create a Neural Network that will be intelligent enough to recognize the characters after training. In this project, we used CNN model for character recognition.

1. For training the model, we'll be using ImageDataGenerator class available in keras to generate some more data using image augmentation techniques like width shift, height shift.
2. Width shift: Accepts a float value denoting by what fraction the image will be shifted left and right.
3. Height shift: Accepts a float value denoting by what fraction the image will be shifted up and down.

For the model, we'll use 4 convolutional layers with a Max pooling layer of window size = (4,4). We'll also use 2 Dense layers where the last dense layers will have 36 output units (26 alphabets + 10 digits) and the activation function used will be 'softmax' because this is a multi-classification problem.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 28, 28, 16)        23248
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 28, 28, 32)        131104
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 28, 28, 64)        131136
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 28, 28, 64)        65600
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 7, 7, 64)          0
_________________________________________________________________
dropout (Dropout)            (None, 7, 7, 64)          0
_________________________________________________________________
flatten (Flatten)            (None, 3136)              0
_________________________________________________________________
dense (Dense)                (None, 128)               401536
_________________________________________________________________
dense_1 (Dense)              (None, 36)                4644
=================================================================
Total params: 757,268
Trainable params: 757,268
Non-trainable params: 0
_________________________________________________________________
```

Fig 9. Model Sequential

The model is now tested and the attribute steps_per_epoch is set to be train_generator.samples, batch size because it ensures the usage of all the train data for one epoch.

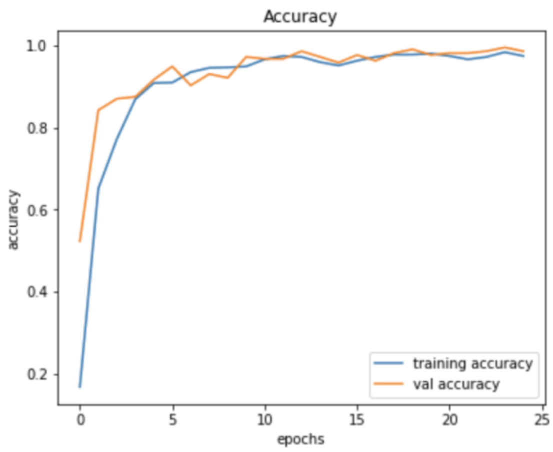

Fig 10. Accuracy graph of trained model

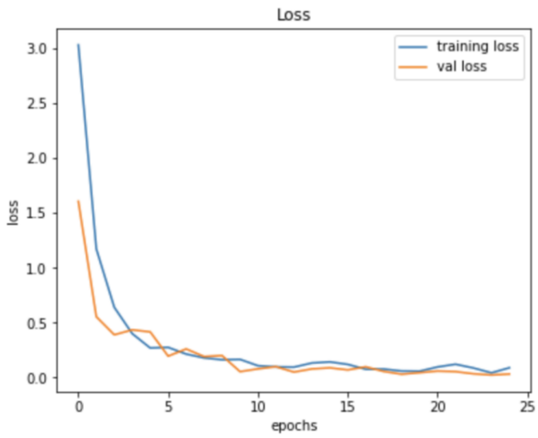

Fig 10. Loss graph of trained model

We now have our license plate and the CNN model ready. We just need to predict each character using the model. For this, we'll first fix the dimension of each character image using the function fix dimension, in which it converts an image to a 3-channel image. The image can then be sent to model.predict_classes() in order to get the predicted character. The dataset is now imported to calculate the overall accuracy of this method. The dataset contains about 200 images with a labels.xlsx that will be useful to check if the predicted output is correct.

Fig 11. Segmented Character and predicted value

7. RESULTS AND CONCLUSION

Automatic number plate recognition is a broad field that can be implemented in a variety of ways. The Automatic Number Plate employed using Contours and convolution neural network (CNN) is our proposed technique. Python was used to complete this implementation, which includes procedures such as localization, segmentation, and recognition. The data set we used contains more than 150 images randomly selected. Finally, the accuracy is calculated for all the car images in the data set and the accuracy that our model obtained was 85.31%.

8. FUTURE WORKS

Automatic number plate recognition (ANPR) can be implemented in a variety of ways. To improve the accuracy, we will also try to implement the same using yolov3 and similar CNN model. We will also try it out using a combined method of Yolo and Contours using the CNN model.

9. REFERENCES

[1] S. Ramalingam, "License plate localization supported morphological operations," 2010 11th International Conference, Singapore, 2010, pp. 1128-1132, DOI: 10.1109/ICARCV.2010.5707933.

[2] J. Yepez and S. Ko, "Improved vehicle plate localization algorithm supported morphological operations," in IET Intelligent Transport Systems, vol. 12, no. 6, pp. 542-549, 8 2018, DOI: 10.1049/ietits. 2017.0224.

[3] M. Kundargi; Automatic car place Recognition Using ANN; IRJET Volume: 02 Issue: 04 | July-2015.

[4] Prasanth Krishna, Pratheesh somarajan; Extraction of auto plate Region in ANPR; international Conference Electrical Technology 2010.

[5] Combined classifier approach for offline handwritten Devanagari character recognition using multiple features." halerao Milind, et al. Computational Vision and Bio-Inspired Computing. Springer, Cham, 2018. 45-54.

[6] Satoshi Urata, Hiroshi Yasukawa, Improvement of contour extraction precision of active contour model with structuring elements, Speech and Signal Processing (ICASSP),IEEE International Conference on Digital Object Identifier, pp.1169-1172, 2012

[7] Kresimir Romic, Irena Galic, Alfonzo Baumgartner, Character recognition based on region pixel concentration for license plate identification, Proceedings of the International Conference on Image Processing (ICIP'94), Vol.2., pp.321-325, 2012.

[8] Xiaoqing Liu and Jagath Samarabandu, Multiscale Edge-Based text extraction from complex images, Multimedia and Expo, 2006 IEEE International Conference, 2006.

[9] Thanongsak,SirithinaphongandKosin Chamnongthai, extracting of car license plate using motor vehicle regulation and character pattern recognition, IEEE international conference of Neural Networks,1998, pp 559-562.

