Chapter 1
A CNN-Based License Plate Recognition Using TensorFlow and PySpark

Lavanya K
Vellore Institute of Technology, India

Bharathi K.
Vellore Institute of Technology, India

Preethi Christina A.
Vellore Institute of Technology, India

Satyam Chaurasia
Vellore Institute of Technology, India

ABSTRACT

The use daily of vehicles is rising exponentially and as a result there is an increase in crimes associated with it. Many vehicles are violating the rules of traffic and so an abnormal number of accidents occur leading to a rise in the crime rates linearly. In order for any vehicle to be recognized, its license plate number is needed. Therefore, the vehicle license plate detection plays a notable role. The optical character recognition (OCR) is one effective way to scan number plates and recognize the text found in the digital image, containing the license plate number into machine readable text which can then be used to track the vehicles. The image of the number plate is first captured, processed, and every character present in the number plate is read for perfect recognition. The optical character recognition model is trained using TensorFlow. Spark's in-memory data engine can perform tasks rapidly in multi-stage jobs. Therefore, TensorFlow and Spark are used together to train and apply the OCR model to perform the license plate recognition swiftly.

DOI: 10.4018/978-1-6684-4246-3.ch001

INTRODUCTION

The increase in the number of vehicles on the road these days is serving as a reason for the increase in traffic and various crimes associated with it. Since the vehicles involved could not be recognized accurately, various cases of theft, hit and run, robbery, kidnapping, smuggling, on-road fatalities remain unsolved. Identifying the vehicles has various applications in toll payments, parking management, road-traffic monitoring, security and crime identification. Manual monitoring of vehicles is cumbersome, error prone and a daunting task. Therefore, a robust mechanism such as an automated vehicle recognition system is necessary to handle this task efficiently.

Each vehicle is uniquely identified by the number assigned to it which is displayed on the license plate. A license plate contains a unique combination of digits and alphabets. When a number from the number plate is correctly detected, the complete information about the vehicle and its owner can be retrieved. One of the effective methods to scan the number plates is optical character recognition.

It involves using OpenCV, Keras. OpenCV stands for *Open Source Computer Vision Library*. It is a library of programming functions that helps with real-time computer vision. Keras is a highly powerful and dynamic framework that makes testing easier, simplifies neural network usage and offers support to both convolution and recurrent networks. There are 3 steps associated with this process: Detection of the plate, character segmentation and reading the contents of the plate. The detection of the plate is done using Tensorflow object detection. TensorFlow is an end-to-end open source platform for machine learning that comprises of tools, libraries and resources that can be used to build and train models using high level APIs and perform powerful experimentation using research. TensorFlow object detection is a computer vision technique that helps in detecting, locating and tracing objects from a video or an image. The model is trained with 411 images of the car with annotated plates. LabelImg, an image annotation tool allows us to annotate images in Pascal VOC format. The dataset is composed of car images that are found online. Some of the images are taken on the street and data augmentation (Vertical Flip, Brightness modification) is performed using Keras. In character segmentation, two methods are used for more accuracy: The first one comprises of a trained model that has images of license plates where characters are annotated. Around 1400 characters are present in total. In the second method, OpenCV's functions are used to process the plate.

At last, to recognize a character, Convolutional Neural Network is trained with Tensorflow and Keras libraries. There are 35 classes (10 for numbers and 25 for alphabet without "O"). Approximately 1000 images are used for each class. A sample of characters and images are collected and data augmentation (rotation and brightness) is performed.