

Getting text from number-plate

Image Processing and Computer Vision Project

Project Work in Image Processing and Computer Vision

Amir Hajiabadi

amir.hajiabadi@studio.unibo.it

Academic year 2021-2022

Abstract

The goal of the project is to develop an application to recognize the number plate in an image and read its number and generate a text output from it.

In the first step, we need to collect images from different resources. Then we manually label each photo of the car, which is an important and time-consuming step.

we do data preprocessing, including normalization, transformation. With the cleared data, we train a powerful Deep Learning model.

We send the part of the photo that we are interested in (ROI), to the Recognition Optical Character to read the text from the photo.

1. Labeling

In the beginning, a folder of images of cars was collected. Then, LabelImg was used. It is an Open-source graphical image annotation tool. It is written in Python and uses Qt for its graphical interface. Annotations are saved as XML files in PASCAL VOC format, the format used by ImageNet. Besides, it also supports YOLO and CreateML formats.

In the README section on GitHub, installation description and instructions are provided.

<https://github.com/tzutalin/labelImg>



Windows + Anaconda

Download and install [Anaconda](#) (Python 3+)

Open the Anaconda Prompt and go to the [labelImg](#) directory

```
conda install pyqt=5
conda install -c anaconda lxml
pyrcc5 -o libs/resources.py resources.qrc
python labelImg.py
```

After running the software, we open the image directory, Set the file format to pascalVOC and click create Rectbox, draw a rectangle around the numberplate, Label it with the name number_plate and save it as an XML file.

Xml to csv:

in the XML files, we are interested in the object element with the actual information about the label. we need to parse this label information and save it into a CSV file called labels.csv

2.Data Processing

● Read Data

We import the required libraries, read the CSV file, put it in a data frame, get the name of each file and its relative address and apply this function to all images.

● Verify Labeled Data

We read the pictures and test the information we have from each picture to assess whether they are correct or not.

● Data Preprocessing

We need some new libraries, separate the values and coordinates of the number plates from the data frame and put them in the labels variable.

We save the shape of each image as h, w, d and divide the coordinates that we had inside each image of the number plate into the first two variables to normalize them.

● Split dataset

We define the train and test variables, And we use train_test_split to divide the data and consider the size as 80%.

3. Deep Learning for Object Detection

● Get Transfer Learning from TensorFlow 2.x

We import several predefined models such as MobileNetV2, InceptionV3, InceptionResNetV2. download, compile, train and save our model.

4. Pipeline Object Detection Model

● Make Predictions

At first, we load our model to make some predictions.

In this section, our output will be normalized.

So, we need to convert them back into what they were originally (denormalization).

● Bounding Box

We use OpenCV to draw Bounding Box on top of our images with the coordinates that we have for every image.

5. Optical Character Recognition (OCR)

● Install Google Tesseract OCR

In this section, we want to install one of the most popular and powerful optical character recognition software used to extract text from the image.

Install tesseract using windows installer available at:

<https://github.com/UB-Mannheim/tesseract/wiki>

Note the tesseract path from the installation. Default installation path at the time of this edit was: C:\Users\USER\AppData\Local\Tesseract-OCR. It may change so please check the installation path.

anaconda prompt:

```
pip install pytesseract  
pip install tesseract  
pip install tesseract-ocr
```

Set the tesseract path in the code before calling image_to_string:

```
pytesseract.pytesseract.tesseract_cmd = r'C:\Users\USER\AppData\Local\Tesseract-  
OCR\tesseract.exe'
```

we must care of two things that can make errors:

Search environment variables in our toolbar search and click on environment variables in opened window, copy our installation path in path variable and our language files path in TESSDATA_PREFIX variable.

● Extract Number Plate Text from Image

Import pytesseract

We define the path of our image and using the object detection function.

we get the image and its coordinates.

Then plt can show our image and its coordinates.

We get the ROI coordinates, then using pytesseract.image_to_string, we extract text from our image.