IEMS5780 Mini-project Report

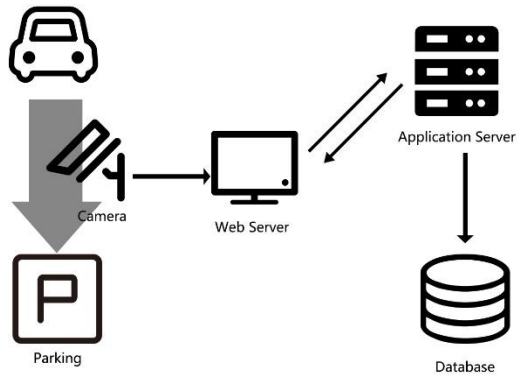
Car Plate Number Recognizing System

Wang Zezhong 1155121439

Ⅰ. Description

My project is deploy a car plate number recognizing model based on deep learning on a Flask server for user to use.The GUI based on Telegram and my server offer services to user via Telegram bot.

This system can deploy with a car parking lot camera together to monitor the car coming in or out automatically the assumption work flow as picture 1 show. But I do not have a camera so I regard the Telegram bot as a camera and when user send a picture to bot, it means a car coming.

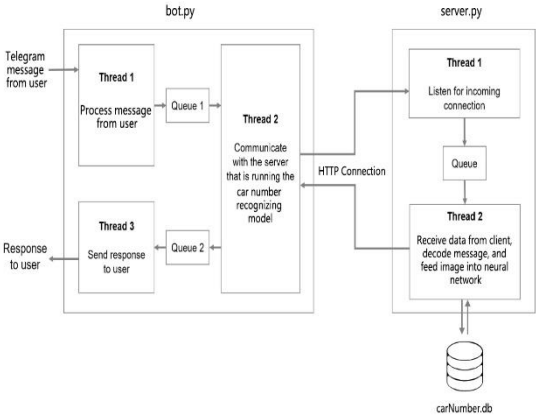


Pic 1 assumption work flow

The server would recognize the car number firstly and the check the number in the database. If there is no same number in the database, we can consider that the car is coming in and insert this number into database. If there is a same number in the database, we can consider that the car is coming out and delete this car's record from the database.

Ⅱ.System Overview

The server and bot communicate with each other by HTTP and the architecture is similar as assignment2. The architecture is as below:



Pic 2 system architecture

Users can communicate with my Telegram bot via any equipment.

Bot.py is responsible for interacting with users. There are 3 threads in this program. One is used to receive users message. The message can be a picture or a pre-defined command. The 2nd thread is used to connect with serve and give the response to thread 3. And thread 3 is responsible for send the result back to users.

Server.py is responsible for process the picture sent by users or operate the database according to the users' command.

carNumber.db is a SQLite datebase. There is just one table in there called number. There are 2 columns in the table, an index which is generated automatically and the other is for storing the car numbers.

Ⅲ.Function

There are 3 API for user to use:

1.User can send my Telegram bot a picture with a Chinese car and the bot would send back the car number to user.

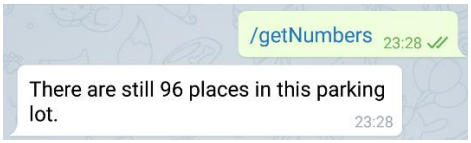
If this car number is the first time seen by the bot, the bot would reply "XXXXX In".

If this car number is the second time seen by the bot, the bot would reply "XXXXX Out".

The screen shot is below:

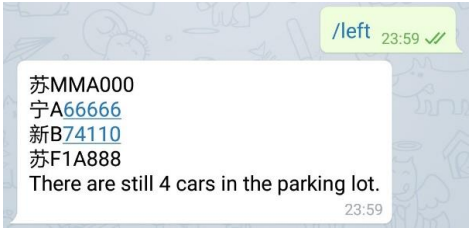
pic 3 demo1

2.User can send "/getnumber" command to check how many place in the car parking available. I assume there are 100 place originally and the bot would send the real-time number back.



pic 4 demo2

3.User can send "/left" command to check which cars are still in the car parking. I design this function for manager but now every one can try it on Telegram.



pic 5 demo3

There are the 3 functions of my bot. As for whether is the car in or out, the server would extract the number out of the picture firstly, and compare the number with the record in the database. So if there are 2 different pictures about a same car, the system still consider them as the same object.

Ⅳ.Model

This model is designed to recognize the car number from a picture and the work flow is as below:

1.Detect the approximate location of the car number by HAAR Cascade, a function in OpenCV library.

2.Extend the location to a rectangle region.

3.Do image binarization and fitting the upper bound and lower bound of the car number by RANSAC.

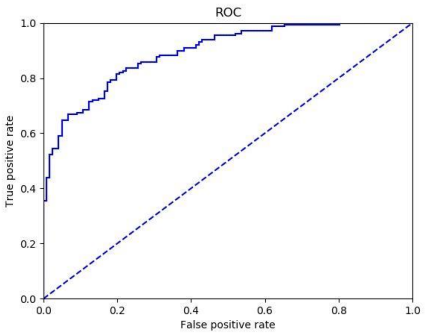
4.Find the left bound and right bound of the car number by CNN Regression.

5.Use CNN sliding window to split numbers and words.

6.Use CNN to recognize numbers and words.

Ⅴ.Performance&Scalability

I value this model by ROC picture.



pic 6 ROC

As for scalability, we can add a cache between application server and database. Because in this system there is a unique character, every car going into the parking must go out. So if we store the number in the caching when car coming in, we can save the time to read and write the database.