



LICENSE PLATE RECOGNITION SYSTEM

Team : Zerone



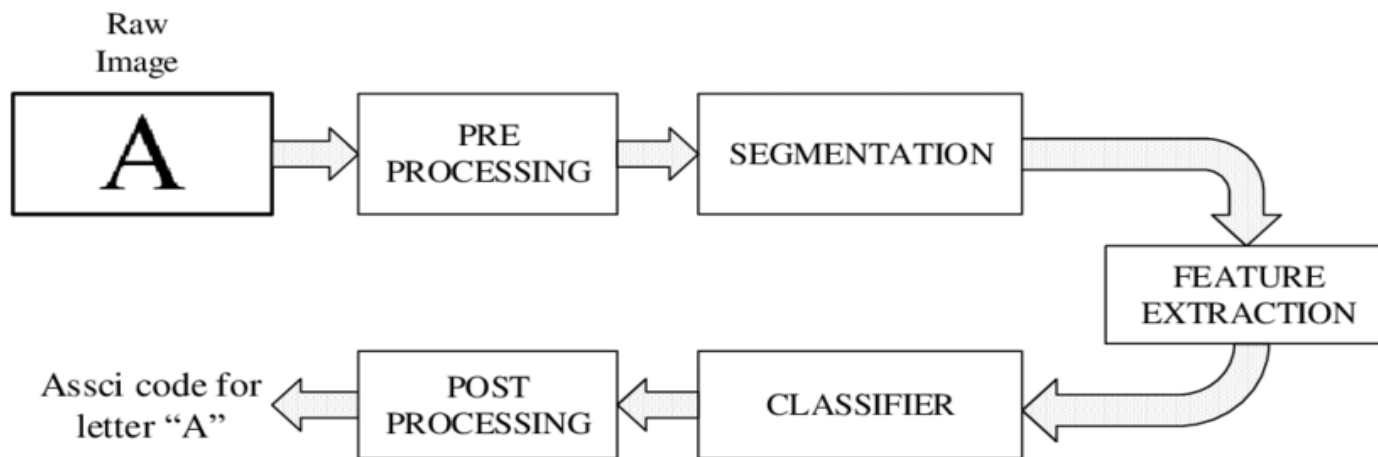
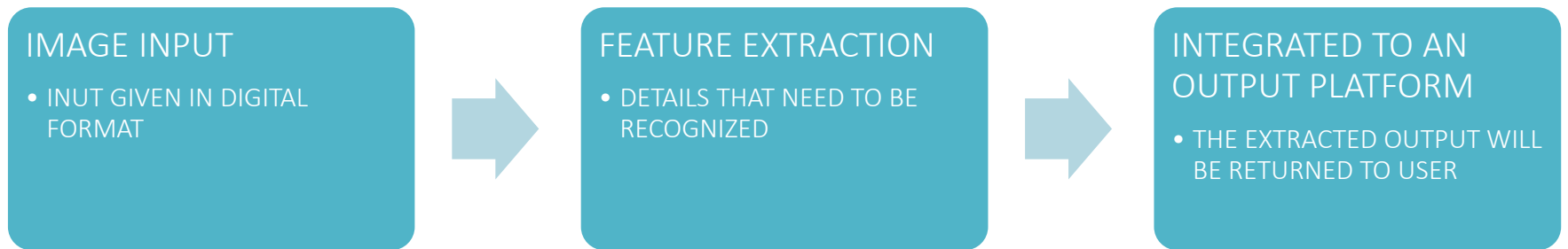
INTRODUCTION

- License Plate Recognition (or '*number plate recognition*') is a special form of optical character recognition (OCR).
- It enables computer systems to read automatically the registration number (license number) of vehicles from digital pictures.
- Reading automatically the registration number means transforming the pixels of the digital image into the ASCII text of the number plate.



BASIC OCR

OCR technology scans paper documents and turn them into electronic, editable files.



NEED IN INDIA?

High Population, has a unique set of needs for ANPR.

To monitor the vehicles' average speed and can identify the vehicles that exceed the speed limit.

Maintain law and order which, in turn, can minimize the number of road casualties.

Cloud-based system for using the license plate data in :

1. Surveillance
2. Management
3. Visual Positioning Systems
4. Automated Toll Collection

Moreover, at an **Individual's and a societal level**, It can be used as a utility or can be integrated with other systems to enhance their functionality

IMPLEMENTATION

SOFTWARE -

Python Libraries and Framework used-

Classifier: *haarcascade* : ML based for code simplification.

OpenCV : for this project in order to identify the license number plates In digital image

Pytesseract : for the characters and digits extraction from the plate.

FRONTEND -

Tkinter, flutter etc

HARDWARE -

A good FPS image capturing device.

BASIC CONCEPT OF LPRS



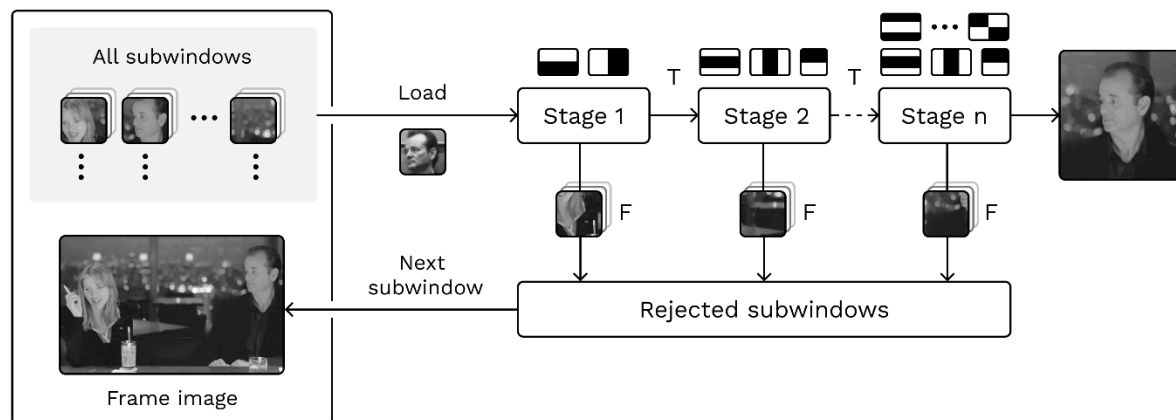
DETECTION

-we have used a classifier - **haarcascade**

haarcascade is a ML based model which is trained on tons of dataset and hence can detect objects from an image. In our case, it will detect number plate from the input image.

-you can use **YOLO** as well, but we have used Haarcascade for simplification of code.

Cascade structure for Haar classifiers



CONVERSION

IMAGE READING = CV2

Open CV computer vision library

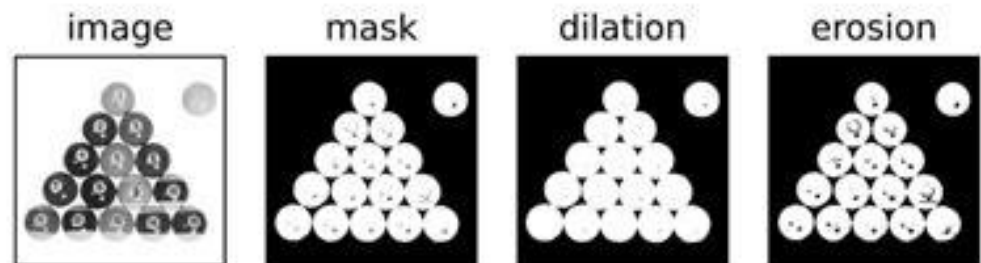
- Most powerful
- contains 1000s of algorithms for identification
- Built in C, C++

we have used techniques like **gray conversion** for converting the input image into grayscale and then performing required operations.

#Cropping number plate ->changing brightness and threshold/sharpening

-using **dilate** and **erode** of cv2 then converting again to b/w

-applying threshold



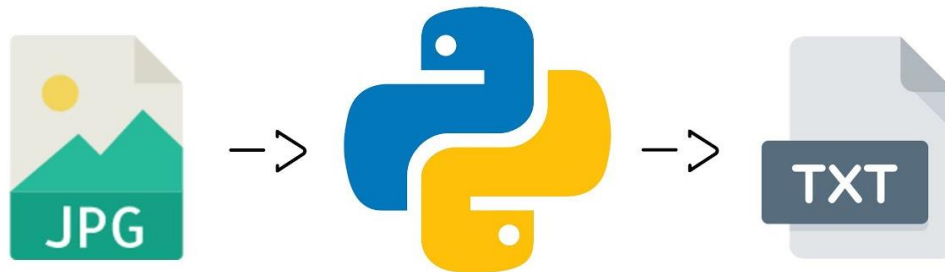
RECOGNITION

- you can use **Deep Learning, CNN** etc.
- we are using pre-trained OCR engine built by open source contributors

PYTESSERACT

- using "**image_to_string**" to convert all content into text
- use string processing to eliminate the extra characters [spaces and extra characters]

Convert IMG to TXT using Pytesseract



File Edit Selection View Go Run Terminal Help Main.py - Untitled (Workspace) - Visual Studio Code

EXPLORER

test images

New

Home

Desktop

Downloads

Music

Pictures

Videos

Documents

This PC

Windows-SSD (C:)

Network

11 items | 1 item selected 173 KB

53

54

55

OUTLINE

TIMELINE

53

54

55

Ln 55, Col 9 Spaces: 4 UTF-8 CRLF Python 3.11.0

```
plate = cv2.erode(plate, kernel, iterations=1)
plate_gray = cv2.cvtColor(plate, cv2.COLOR_RGB2GRAY)
thresh, plate = cv2.threshold(plate_gray, 127, 255, cv2.THRESH_BINARY_INV)

pytesseract.image_to_string(plate)
read = ''.join(e for e in read if e.isalnum()) #only alphanumeric character will print alnum=alphanumeric
("The Number Plate is :\n", read)

rectangle(img, (x, y), (x+w, y+h), (51, 51, 255), 2)
rectangle(img, (x, y - 40), (x+w, y), (51, 51, 255), -1)
scale = 1
color = (255, 255, 255) #white color
thickness = 2
cv2.putText(img, read, (x, y-10), cv2.FONT_HERSHEY_SIMPLEX, fontScale, color, thickness, cv2.LINE_AA, False)

cv2.imshow('Scanned Plate', plate)

cv2.imshow("Result", img)
cv2.imwrite('result.jpg', img)
cv2.waitKey(0)
cv2.destroyAllWindows()

./test_images/p5.jpg')
```

Input Image [Still Frame or from device]

The screenshot illustrates the ANPR System interface and its output. The main window, titled "ANPR System", features a "Select Image" button (highlighted with a red box) and a "Perform ANPR" button (also highlighted). A red arrow points from the "Select Image" button to a "Select Image" dialog box. This dialog shows a file explorer view of a folder named "test i..." containing several image thumbnails labeled p1 through p8. The thumbnail labeled p5, which shows the rear of a red car, is selected and highlighted with a red box. Another red arrow points from the "Perform ANPR" button to the "Result" window. The "Result" window displays a large image of the red car with a red bounding box around the license plate, which reads "DL7CN5617". A small "Scanned..." window also shows the extracted license plate text "DL7CN5617". A red arrow points from this "Scanned..." window to a black box at the bottom left that contains the text "The Number Plate is : DL7CN5617".

ANPR System

Select Image

Select Image

File name: .jpeg files

Open Cancel

ANPR System > Main.py > extract_num
FROM pytesseract

PROBLEMS OUTPUT DEBUG CONSOLE

PS C:\Users\ARYAN\OneDrive\Desktop\ANPR System> extract_num
The Number Plate is :
DL7CN5617

Scanned... DL7CN5617

Perform ANPR

The Number Plate is :
DL7CN5617

DL7CN5617

DL7CN5617

Team-BHP.com

Extracted Plate as "Text" / For Management and easy operations in DBMS

APPLICATIONS

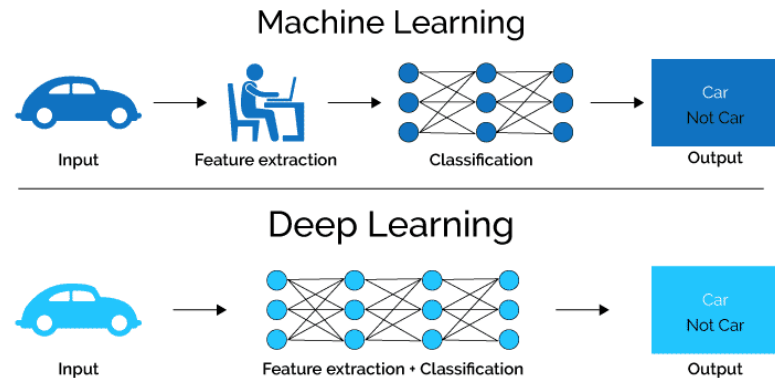
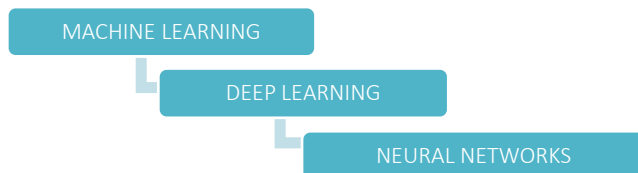
- Intelligent Transportation Systems (ITS),
- security systems
- motorway toll collection (highway toll collection),
- traffic analysis,
- police law enforcement,
- vehicle theft prevention,
- security monitoring of roads and checkpoints
- vehicle surveillance

INNOVATION + CONCLUSION

FUTURE ASPECTS

The idea of OCR is basic, but its actual potential is yet to be discovered. Contributing to the same, we will try to implement it on real-time applications:

- Multiple Inputs
- Image/Video Capture
- As a utility for software
- Management & Surveillance



REFERENCES

www.anpr.net

en.wikipedia.org

<https://www.javatpoint.com>

<https://platerecognizer.com/anpr-for-india/>

THANK you