AUTOMATIC NUMBER PLATE RECOGNITION

A PROJECT REPORT

Submitted by

Hari Haran. N	**510617105004**
Pradeep Kumar. B	**510617105020**
Shyam Kumar. S	**510617105026**
Vishnu Varadhan. A	**510617105032**

In partial fulfilment for the award of the degree

Of

BACHELOR OF ENGINEERING

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

C.ABDUL HAKEEM COLLEGE OF ENGINEERING AND TECHNOLOGY, MELVISHARAM, RANIPET-632509

ANNA UNIVERSITY: CHENNAI 600 025

APRIL 2021.

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"AUTOMATIC NUMBER PLATE RECOGNITION SYSTEM"** is the bonafide work of **"Hari Haran. N (510617105004), Pradeep Kumar. B (510617105020), Shyam Kumar. S (510617105026), Vishnu Varadhan. A (510617105032).** Who carried out the project work under my supervision .

SIGNATURE
Mohamed Hashim. C, B.E., M.S.,

HEAD OF THE DEPARTMENT
Electrical and Electronics Engineering
C.Abdul Hakeem College of Engineering & Technology,
Melvisharam-632 509,
Ranipet District.

SIGNATURE
Mohamed Hashim. C, B.E., M.S.,
Supervisor

HEAD OF THE DEPARTMENT
Electrical and Electronics Engineering
C.Abdul Hakeem College of Engineering & Technology,
Melvisharam-632 509,
Ranipet District.

CERTIFICATE OF EVALUVATION

College Name : C. Abdul .Hakeem College of Engineering And Technology, Melvisharam.

Branch : Electrical and Electronics Engineering

Semester : VIII

Title of the Project **:** FACE DETECTION AND RECOGNITION USING JETSON NANO

S.NO	NAME OF STUDENTS	REGISTER NO
1	Hari Haran. N	510617105004
2	Pradeep Kumar. B	510617105020
3	Shyam Kumar. S	510617105026
4	VishnuVaradhan. A	510617105032

Name of the supervisor: **Mr. Mohamed Hashim. C, B.E., M.S.,HEAD OF THE DEPARTMENT OF EEE**

The report of the project work submitted by the above students in the partial fulfilment for the award of bachelor of engineering degree in the **ELECTRICAL AND ELECTRONICS ENGINEERING** of Anna university were evaluated and confirmed to be report of the work done by the above students.

Submitted for the university examination held on ……

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGMENT

First and foremost, we thank to the ALMIGHTY for his substantial blesses and mercy at all stages in the completion of this project work.

We express our profound gratitude to our honorable chairman **JANAB HAJI S.ZIAUDDIN AHMED SAHIB,** and to our beloved correspondent **HAJI V.M.ABDUL LATHEEF SAHIB,** for giving us the opportunity to serve the purpose of education.

We are highly indebted to our principal **Dr.J.BASKARAN** for his valuable suggestion and guidance. Our sincere gratitude to our Head of the Department **Mr. MOHAMMED HASHIM B.E, M.S,**

We express our deepest sense of gratitude and revertible thanks to our guide **Mr. MOHAMMED HASHIM., B.E M.S,** for his consecutive encouragement and who has been so helpful in all aspect and for the admirable facilities provided to us during the course of the project work.

We greatly indebted to all our beloved department faculty members for this constant inspiration, support, encouragement and guidance right from the conception to the completion of our project.

We also acknowledge the help and support rendered by all our department non-teaching staffs in completion of our project. We convey our heartiest thanks to our beloved parents who helped us to complete this herculean task, which is assigned to us. Without them, this could not have seen the light of the day. The light of sprit will always burn in us because they nourished so mercifully.

ABSTRACT

This thesis develops an algorithm for automatic number plate recognition (ANPR) and implemented it on the NVIDIA Jetson Nano. It is an image processing technology which uses number plate to identify the vehicle. The objective is to design an efficient automatic authorized vehicle identification system by using the vehicle number plate.

It can be used to implement on the entrance for security control of a highly restricted area like military zones or area around top government offices e.g. Parliament, Supreme Court etc. The algorithm developed here is aimed to be lightweight so that it can be run in real time. With images provided by a USB web camera the system will be able to recognize the number plate under normal condition.

The algorithm is built in three sections; the first is the initial detection of a number plate using edge and intensity features in the image; in the second, the text of the number plate is found; last is the actual character recognition.

Optical character recognition technique is used for the character recognition. The resulting data is then used to compare with the records on a database. The major advantages of the system is its real-time capability and that it does not require any additional sensor input (e.g. from infrared sensors) except a video stream. The system is implemented in python, and its performance is tested on real image. It is observed that the developed system successfully detects and recognize the vehicle number plate on real images.

TABLE OF CONTENT

LIST OF FIGURE

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Automation is the most frequently spelled term in the field of electronics. Due to automation, revolution has occurred in the existing technologies. Automatic Number Plate recognition is a very secure and accurate system that is used for reading the number plate of the vehicles without involving human interaction. In last few years, ANPR or license plate recognition (LPR) has been one of the useful approaches for vehicle surveillance. It can be applied at number of public places for fulfilling some of the purposes like traffic safety enforcement, automatic toll text collection, car park system and Automatic vehicle parking system. ANPR algorithms are generally divided in four steps: Vehicle image capture Number plate detection Character segmentation and Character recognition. It makes use of high-speed image capturing with supporting illumination. This is very useful for automating toll booths, automated signal breakers identification and finding out traffic rule breakers. It is a technology that uses optical character recognition on images to read vehicle registration plate. Automatic number-plate recognition can be used to store the images captured by the cameras as well as the text from the license plate, with some updates it can also store a photograph of the driver. Systems commonly use infrared lighting to allow the camera to take the picture at any time of day or night.

1.2. NEED

ANPR is used for Detecting crime through the use of intelligence monitoring. Identifying stolen vehicles, Detecting vehicle document crime, Electronic toll collection etc., It can be further developed to detect traffic violations

1.3. STATE DIAGRAM:

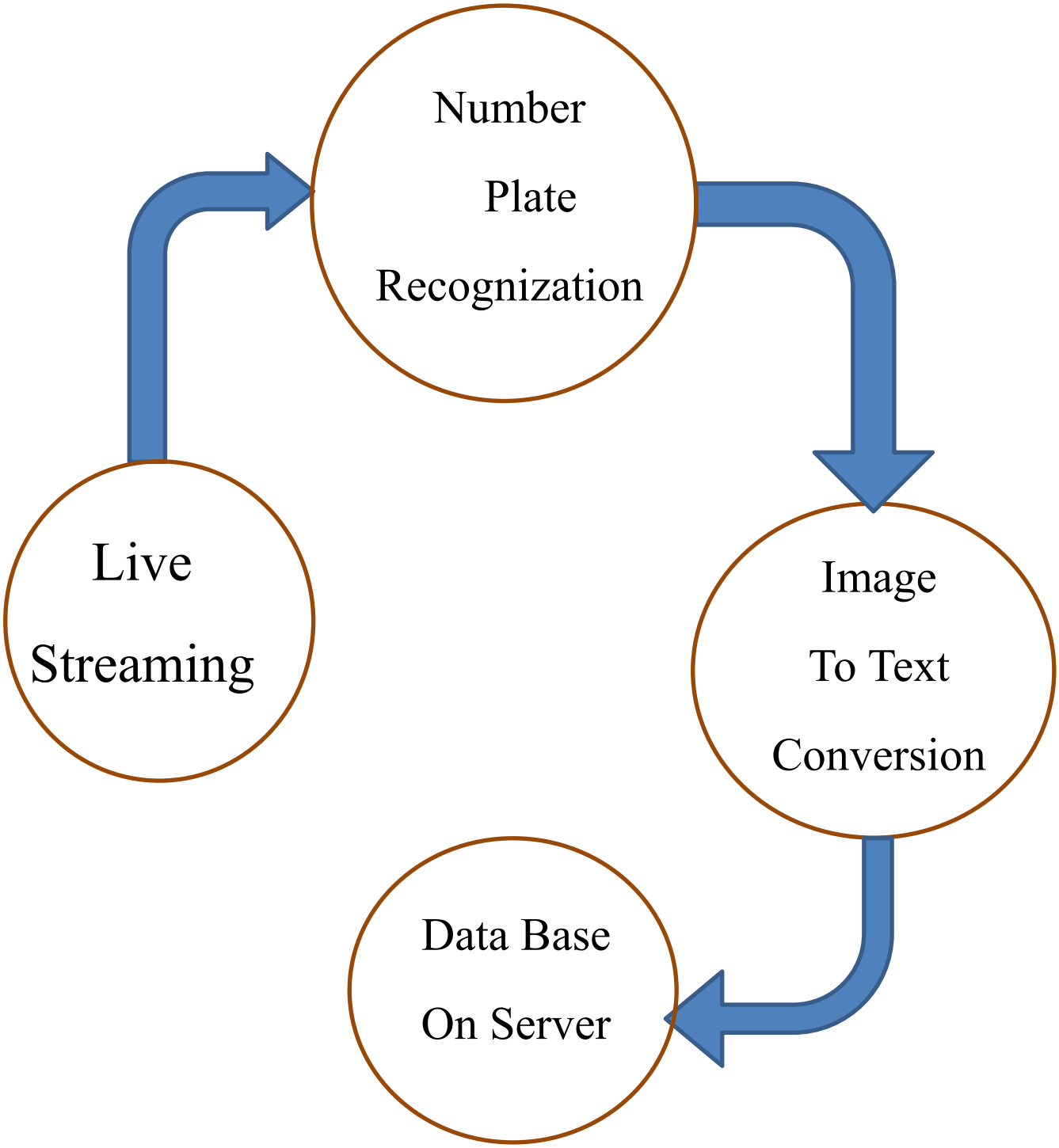


Fig 1.1 state diagram

1.4. SCOPE

There is no proper device which is very much faster and accurate. As per the study conducted by Market Research Future (MRFR), the global automatic number plate recognition (ANPR) system market is expected to surpass a valuation of USD 1,100 million by the year 2023. Enforcement of stricter traffic laws and increased focus on public safety is driving the adoption of ANPR systems. They offer cost effective and credible surveillance solutions in public vicinities. These systems are expected to witness a strong demand in the foreseeable future.

1.6. SUMMARY

Traffic control and vehicle owner identification has become major problem in every country. Sometimes it becomes difficult to identify vehicle owner who violates traffic rules and drives too fast. Therefore, it is not possible to catch and punish those kinds of people because the traffic personal might not be able to retrieve vehicle number from the moving vehicle because of the speed of the vehicle. Therefore, there is a need to develop Automatic Number Plate Recognition (ANPR) system as a one of the solutions to this problem. This ANPR system is secure and accurate one. If this system is employed in our society then crimes can be minimized and records can be maintained. We can expect this system to create a larger impact in our environment.

CHAPTER 2

LITERATURE SURVEY

2.1 ANPR system on an ARM-DSP

This complete system, is an embedded standalone, intelligent and capable of capturing and processing license plates on board the device, and represents an advance on the traditional commercial ANPR system which uses a standard definition camera to capture the vehicles, with a separate nearby computer to process the images. The major advantages of the embedded system presented here include a reduction in cost and increased portability, as the system no longer requires separate processing hardware and expensive multiple data transferring media. The algorithm developed for the ANPR system is optimised for the multicore DM8168 Chip from Texas Instruments, which contains ARM CORTEX - A8 (1.2 GHz) and C674x DSP (I GHz) which is a floating-point Very-Long-Instruction-Word (VLlW) DSP. The optimisation is achieved through the efficient use of internal and external device memory with different resources utilised depending on its suitability for the algorithm used, in order to achieve real time processing. In addition, we have taken advantage of the HD video processing subsystem on the device to facilitate full HD video capture (1920 X 1080), encoding and decoding. The video frames are then transferred to the ARM side of the Chip for processing; which is where communication with the DSP is established. The overall system operates at 14 frames per second and with over 95% recognition success using a large (70K plus) UK and European database of LP images. The ANPR algorithms developed are advanced work based on earlier research work using fixed point DSPs published by the authors in [2], [3] and [4].

2.2ANPR on FPGA

The second part of the investigation focused on developing and accelerating a full ANPR algorithm on FPGA. A range of image processing algorithms and architectures for each ANPR stage (i.e. NPL, NPS and OCR) have been developed and optimised to exploit features and innovations available within new FPGAs [5][6][7]. The proposed architectures have been implemented and verified using the Mentor Graphics RC240 FPGA development board equipped with a 4M Gates Xilinx Virtex-4 LX40. The ANPR full algorithm takes less than 10 ms and consumes only 80% of the FPGA on-chip resources. The overall results achieved show that the entire ANPR algorithm can be implemented on a single FPGA that can be placed within an ANPR camera housing to create a stand alone unit.

Limitations:

The ARM-DSP based ANPR system described is designed for commercial applications where the need for low power, low prices and real time systems is vital. A single FPGA can also be added as a "plug-in" to the ARM-DSP based hardware SoC, depending on the extra resources needed for the application. The overall results have shown that it is possible to use cheaper off-the-shelf ARM-DSPs and FPGAs multicore processors for "standalone" ANPR systems through device and algorithm optimisation to achieve real-time performance at higher recognition rate using efficient algorithms.

* Corresponding author: Hongsong Zhu, Email: zhuhongsong@iie.ac.cn. less studied. How to detect sudden acceleration or cardiac arrest behavior near checkpoints or important institutions are studied in [9]. Vision-based anomaly detection schemes are usually distributed, which independently process data collected from a few roads. The distributed nature of those schemes makes it hard to continuously track vehicle objects and collaboratively detect anomalies from the data collected from thousands cameras.

Trajectory-based data mining techniquesTrajectory-based data mining techniques first extract spatial and temporal information from trajectories, and then analyze the vehicle behavioral patterns. To collect vehicle trajectory data, many research activities exploited the geo-location information provided by on-board GPS devices. For instance, estimating the speed of taxis is studied in [10]; [11] used GPS information to study vehicle density; [12] used GPS positioning data to detect vehicle speeding behaviors. Exploiting GPS data to detect vehicle anomalies has good precision. However, there are considerable overhead in installing the additional positioning and communicating devices on every vehicle, and later collecting data via networks.

The successful of the number plate recognition is depend on the correctly of the As a component of an ANPR system, successful character recognition on a number plate is to correctly specify the each component of the number plate either as the character on the number plate from A to Z or as a digit number from 0 to 9. In this section, the existing methods which are handles the each of process of a character recognition system will be presented. In addition, the reviews of some well-known method for character recognition will discuss in this section. The common processes of ANPR are consists of following five steps. There are image acquisition, pre-processing, feature extraction, segmentation, and recognition as shown in

A. Image Acuisition

Image acquisition is first important step for the ANPR system. It is due to this step offers the input image for the whole process in the ANPR system. The previous researchers used many capture method. In paper published by Maarif and Sardy [5], they used digital camera to acquire license plate. The infrared lighting that embedded in camera was presented in Badr et.al. paper [6]. In another paper published by Kim et. Al. [7] and Yan et.al. [8] were presented a video camera to capture the number plate image.

B. Image Pre-processing

The objective of this step is to increase the image quality especially to sharpen the edge features of the image. Numerous methods were proposed by many researchers for the preprocessing step. For example, Khalifa et. al. [9] enhance the input image by convert it to gray scale by taking the luminance component of National Television System Committee (NTSC) standard method. Duan et. al. [10] used detection method based on the boundary features. The method sequentially consists of graying the image, normalizing the image using Sobel filters to extract the edging image and histogram equalization using adaptive thresholding algorithm to binarize the image. Finally, Yo-Ping Huang et. al. [11] used Gaussian filter in order to smooth the image for reduction influences noises. It will make the pixels near the center of image have higher value than the pixels far for the center of the image. To enhance the contrast of the image, they used Power Law transformation.

C. Feature Extraction

In humans' perception, a number plate is a small of plastic or metal plate patch to a vehicle for official identification purposes. However, the machines do not understand this description as well as they do not understand what vehicle, or whatever else is. In consequence, the alternative way of description of a number plate is needed by the machines. Kim et. al. [7] in their paper showed the alternative way to teach the machine what and where is the number plate. The methods they used for this stage are two Neural Network-based filters as an input and a post processor to combine the two filters images in order to locate the license plate. The filter that used in two Neural Networks are vertical and horizontal filters, which observe small windows of vertical and horizontal cross sections of an image and resolve whether each window contains a license plate. Cross-sections have adequate data for differentiating a plate from the background. Duan et. al. [10] proposed the combination of the Hough

Transform and Contour algorithm that produces higher accuracy and fast processing, thus can be implemented into real time system. While Badr et. al. [6] were applying Sobel vertical edge detection, threshold and doing a vertical projection (Y axis) in sequent. The process will produce the strongest peak. The image will cut according the range of this peak.

D. Character Segmentation

Many different methods for character segmentation have been proposed in the previous researches and some of them are as follows. Khalifa et. al. [9] used Connected-Component which is following the four steps to segment the character of the license plate. The first step is stretching the contrast of the image to extend over the entire range of gray levels available (0-225). The next step is threshold the image using Otsu method. Thirdly, search for connected components in the image and assign a special label for each connected component. Finally, resize each character from previous step to the standard size (20x10) to be used for next step recognition process. Duan et. al [10] used a horizontal projection for detecting and segmenting rows in two rows plates and vertical projection for character segmentation. While, Maarif and Sardi [5] used thin window scanning with the size of partition 56 x 1 pixels. They divided the plate into three blocks. First, block contained letter which corresponds to area domain of the car. Second, block contains number and the last is block contained letter.

E. Optical Character Recognition

Optical character recognition (OCR) is the process to convert the images of handwritten, or typewritten into machine encoded text. In the previous researches, there are numerous methods such as Euclidean distance [12], Hidden Markov Model (HMM) [10], Artificial Neural Network (ANN) [13] [14] [9], Support Vector Machine (SVM) [15] and template matching [16] [17].

CHAPTER 3

HARDWARE REQUIREMENTS

3.1 GENERAL

In this chapter, we have discussed about features of each components used in this project.

3.2 COMPONENTS USED

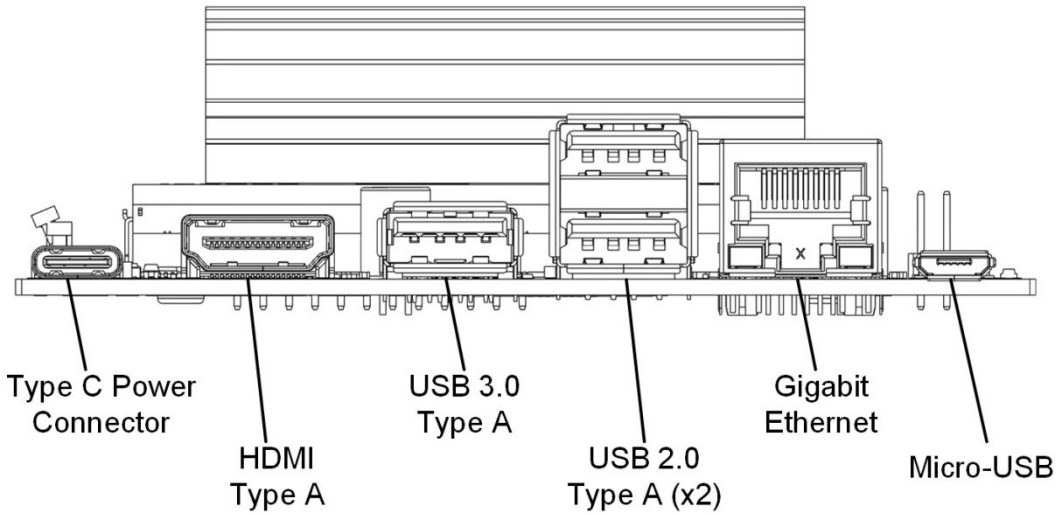
The main components used here are,

NVIDIA Jetson nano	2GB RAM
Webcamera	C270
Memory card	64GB
Monitor	HDMI TYPE
Adapter	3A/5V

3.2.1 JETSON NANO 2GB

It's small, powerful, and priced for everyone at a much lower price. This means educators, students, and other enthusiasts can now easily create projects with fast and efficient AI using the entire GPU-accelerated NVIDIA software stack.All these resources are enabled by NVIDIA JetPack, which brings to each Jetson developer the same CUDA-X software and tools used by professionals around the world. JetPack includes a familiar Linux environment and simplifies the development process with support for cloud-native technologies such as containerization and orchestration. The Jetson Nano 2GB Developer Kit delivers incredible AI performance at a low price. It makes the world of AI and robotics accessible to everyone with the exact same software and tools used to create breakthrough AI products across all industries.

Front View



Rear View

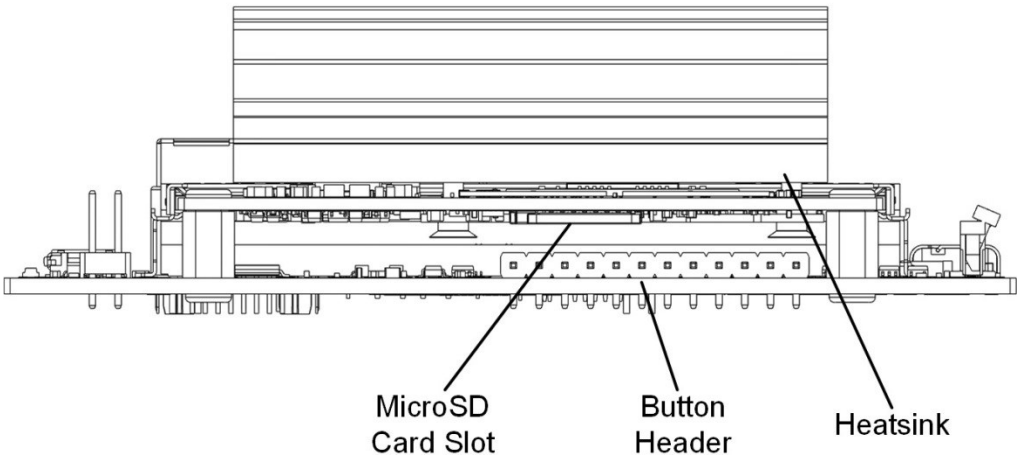


Fig:3.1JETSON NANO 2GB

3.2.2 WEB CAMERA

A webcam is a video camera that feeds or streams an image or video in real time to or through a computer to a computer network, such as the Internet. Webcams are typically small cameras that sit on a desk, attach to a user's monitor, or are built into the hardware. Webcams can be used during a video chat session involving two or more people, with conversations that include live audio and video.Webcam software enables users to record a video or stream the video on the Internet. As video streaming over the Internet requires much bandwidth, such streams usually use compressed formats. The maximum resolution of a webcam is also lower than most handheld video cameras, as higher resolutions would be reduced during transmission. The lower resolution enables webcams to be relatively inexpensive compared to most video cameras, but the effect is adequate for video chat sessions. A webcam is a video camera that feeds or streams an image or video in real time to or through a computer to a computer network, such as the Internet. Webcams are typically small cameras that sit on a desk, attach to a user's monitor, or are built into the hardware. Webcams can be used during a video chat session involving two or more people, with conversations that include live audio and video.Webcam software enables users to record a video or stream the video on the Internet.

Fig: 3.2WEB CAMERA

3.2.3 MEMORY CARD 64GB

A Memory card or memory cartridge is an electronic data storage device used for storing digital information, typically using flash memory. These are commonly used in portable electronic devices, such as digital cameras, mobile phones, laptop computers, tablets, PDAs, portable media players, video consoles, synthesizers, electronic keyboards and digital pianos, and allow adding memory to such devices without compromising ergonomy, as the card is usually contained within the device rather than protruding like USB flash drives.

Fig: 3.3 MEMORY CARD 64GB

3.2.4 MONITOR

A computer monitor is an output device that displays information in pictorial form. A monitor usually comprises the visual display, circuitry, casing, and power supply. The display device in modern monitors is typically a thin film transistor liquid crystal display (TFT-LCD) with LED backlighting having replaced cold-cathode fluorescent lamp (CCFL) backlighting.Monitors are connected to the computer via HDMI connectors and signals. Originally, computer monitors were used for data processing while television sets were used for entertainment. From the 1980s onwards, computers (and their monitors) have been used for both data processing and entertainment, while televisions have implemented some computer functionality.

Fig: 3.4 MONITOR

3.2.5 ADAPTER

An AC adapter, AC/DC adapter, or AC/DC converter is a type of external power supply, often enclosed in a case similar to an AC plug. Other common names include plug pack, plug-in adapter, adapter block, domestic mains adapter, line power adapter, wall wart, power brick, and power adapter. Adapters for battery-powered equipment may be described as chargers or rechargers .AC adapters are used with electrical devices that require power but do not contain internal components to derive the required voltage and power from mains power. The internal circuitry of an external power supply is very similar to the design that would be used for a built-in or internal supply.. 5V/3A OUTPUT

Fig: 3.5 ADAPTER

CHAPTER 4

SOFTWARE REQUIREMENTS

4.1 GENERAL

In this chapter, we have discussed about Software and packages used in this project.

4.2 SOFTWARE

1. OS JETPACK 4.4.1(UBUNTU LX11) AN LINUX based operating system
2. PYTHON as programming language

4.2.1. Installing OS on SD card

1. Write Image (OS) to the micro SD Card

To prepare your micro SD card, you'll need a computer with Internet connection and the ability to read and write SD cards, either via a built-in SD card slot or adapter.

1. Download the Jetson Nano 2GB Developer Kit SD Card Image and note where it was saved on the computer.
2. Write the image to your micro SD card by following the instructions below according to the type of computer you are using: Windows

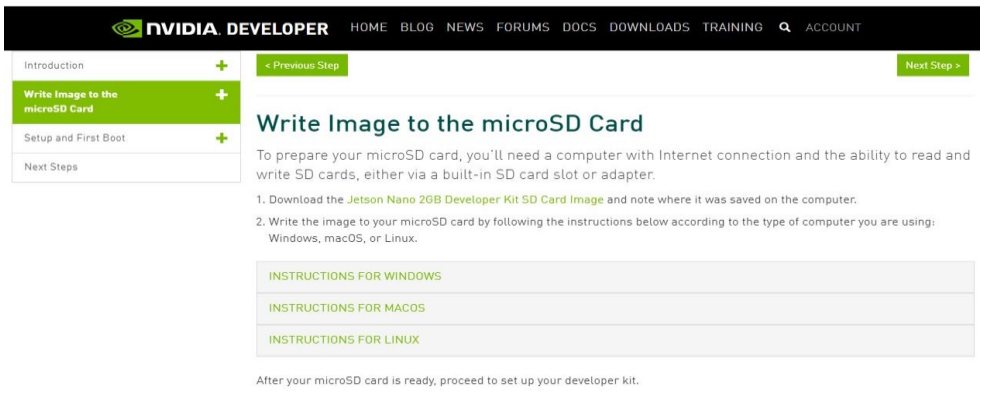


FIG 4.1 nvidia website

2. Format your micro SD card using SD Memory Card Formatter from the SD Association.

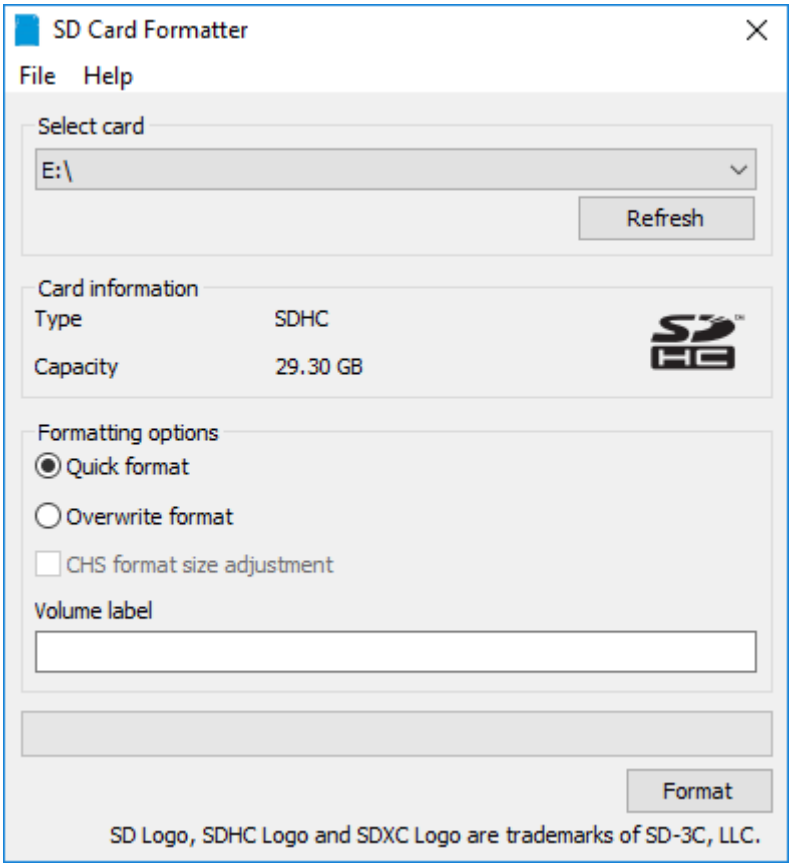


FIG 4.2formatting SD card

Download, install, and launch SD Memory Card Formatter for Windows.

1. Select card drive
2. Select “Quick format”
3. Leave “Volume label” blank
4. Click “Format” to start formatting, and “Yes” on the warning dialog

3. Use Etcher to write the Jetson Nano Developer Kit SD Card Image to your micro SD card

1. Download, install, and launch Etcher.
2. Click “Select image” and choose the zipped image file downloaded earlier.

3. Insert your micro SD card if not already inserted.

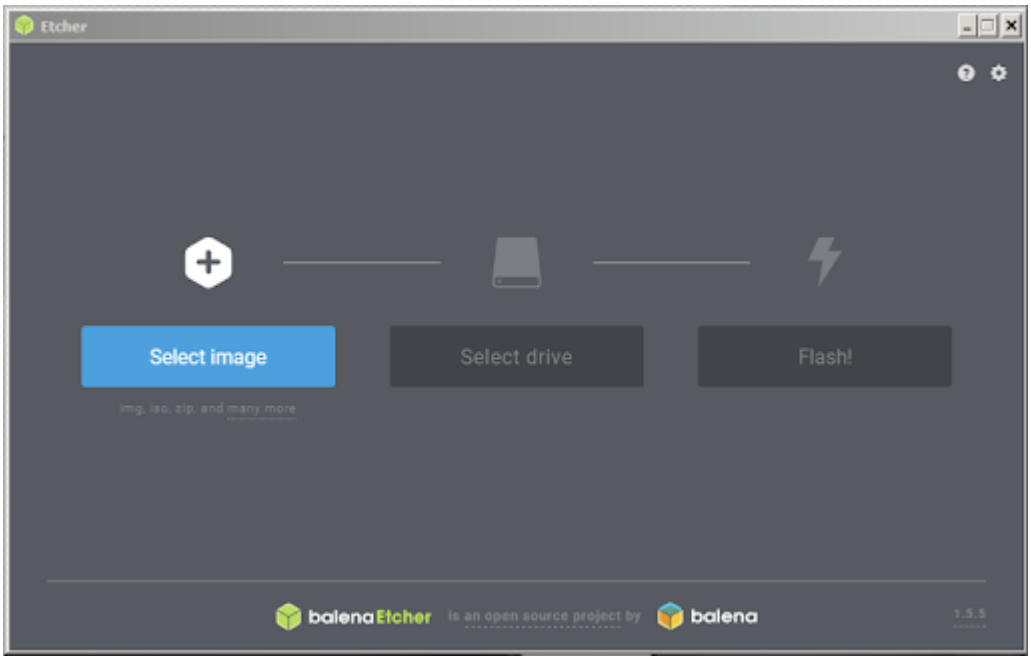


FIG 4.3selct image file

4. Click “Flash!” It will take Etcher about 10 minutes to write and validate the image if your microSD card is connected via USB3.

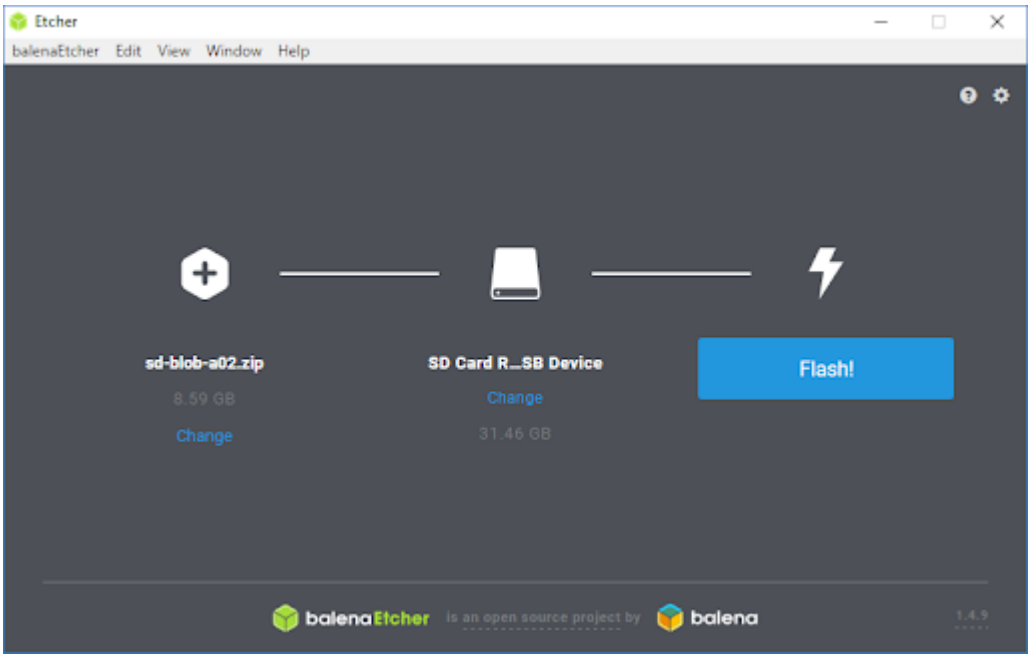


FIG 4.4 select sd card and flash

5. After Etcher finishes, Windows may let you know it doesn’t know how to read the SD Card. Just click Cancel and remove the microSD card.

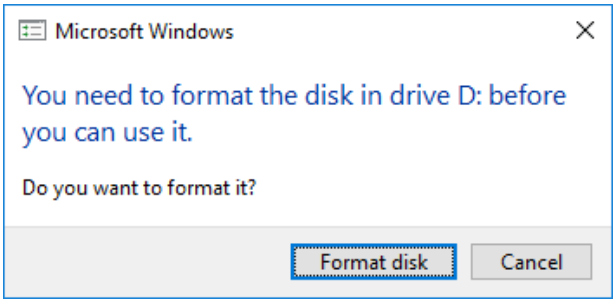


FIG 4.6formatte SD card

6. After your microSD card is ready, proceed to set up your developer kit.

Eject your sd card form the computer And insert into jetson nao

4.2.2. Python

This process uses the **apt package manager** to install Python. There are fewer steps, but it's dependent on a third party hosting software updates. You may not see new releases as quickly on a third-party repository.

Most factory versions of Ubuntu 18.04 or Ubuntu 20.04 come with Python pre-installed. Check your version of Python by entering the following:

$ python —version

If the revision level is lower than 3.7.x, or if Python is not installed, continue to the next step.

Step 1: Update and Refresh Repository Lists

Open a terminal window, and enter the following:

$ sudo apt update

Step 2: Install Supporting Software

The software-properties-common package gives you better control over your package manager by letting you add PPA (Personal Package Archive)

repositories. Install the supporting software with the command: $ sudo apt install software-properties-common

Install Python 3

Now you can start the installation of Python 3.8 with the command:

$sudo apt install python3.8 Allow the process to complete and verify the Python version was installed sucessfully.

$ python —version

4.2.2.1Packages for python

4.2.2.2INSTALLING JETSON INFERENCE

Cloning Jetson inference:

$ git clone --recursive https://github.com/dusty-nv/jetson-inference

Open Jetsin inference -> $ cd jetson-inference

Then Run the Docker container to download and install the models ->

$ docker/run.sh

Then model downloader will appear you have to download (32. FCN-Resnet18-Pascal-VOC-512*320) if your network speed is good than it will take 30 or less than 30 minutes. Because it will start to download the Docker file called dustynv/jetson-inference:r32.4.4(900 MB) for Jetpack 4.4.1 version and L4T R32.4.4 version.

Version based Docker Files:

Container Tag	L4T version	JetPack version
dustynv/jetson-inference:r32.5.0	L4T R32.5.0	JetPack 4.5
dustynv/jetson-inference:r32.4.4	L4T R32.4.4	JetPack 4.4.1

dustynv/jetson-inference:r32.4.3	L4T R32.4.3	JetPack 4.4

you can check whether it is installed correctly by this commands:

$ cd jetson-inference

$ docker/run.sh

Pic 1

Then you have to download some other models for Real-time ANPR system:

$ cd jetson-inference/tools

$ ./download.model.sh

Download following model packages -> 1. SSD-Mobilenet-v1

4.2.2.3 INSTALLING PIL

Now we are ready to install PIL. Type the following:

$ sudo pip install pil

To install Pillow (recommended), type the following:

$ sudo pip install Pillow

PIL (or Pillow) should now install with support for JPEGs, PNGs and FreeType

3.2.2.4 INSTALLING PYTESSERACT

Now we are ready to install pytesseract . Type the following.

$ pip install pytesseract

3.2.2.5 INSTALLING OPENCV

OpenCV can be directly downloaded and installed with the use of pip (package manager). To install OpenCV, just go to the terminal and type the following command:

$ pip3 install opencv-python

3.2.2.6 INSTALLING NUMPY

The recommended approach is to install the stable Numpy module directly from the Ubuntu repository:

$ sudo apt install python-numpy

CHAPTER 5
METHODOLOGY

5.1 General

In this chapter, we have described about the tools or software or hardware that we have implemented on our project. With the usage of these, we have made this project simple and equipped with latest technologies.

5.2 Methodology of our project

In this chapter the methodology of our project is described in details. First, we capture the image of a number plate by an camera(Web Camera, CCTV camera or Raspberry Pi camera or CSI camera). Then we Process the image to get the characters in that saved image of the number plate.

5.2.1 Deep Learning Inference

A Deep Learning model consists of a neural network with internal parameters, or weights, configured to map inputs to outputs. In Image Classification, the inputs are the pixels from a camera image and the outputs are the possible categories, or classes that the model is trained to recognize. The choices might be 1000 different objects, or only two. Multiple labeled examples must be provided to the model over and over to train it to recognize the images. Once the model is trained, it can be run on live data and provide results in real time. This is called inference.

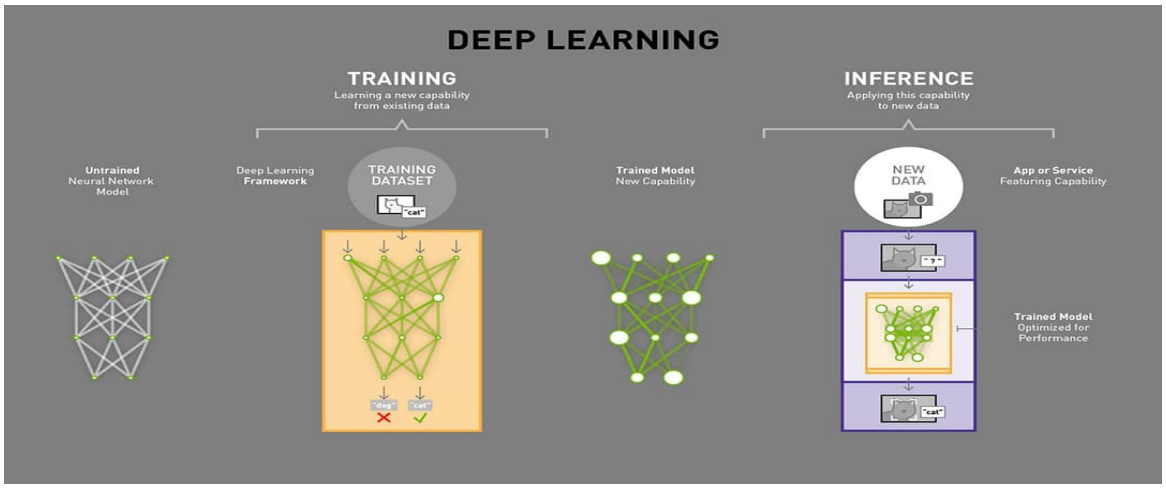


FIG 5.1 structure

5.2.2 Creating Datasets

A data set (or dataset) is a collection of data. Here we utilize images as a data .We collected images of various number plates. Each value is known as a datum. Data sets can also consist of a collection of documents or files.

FIG 5.2 number plate

5.2.3 Annotating Images

For annotating collected images we used LABELIMG graphical annotation tool Labelimg also used in windows environment.

FIG 5.3 labeling number plate

The images are annotated in PASCAL VOC format. After creating this will create a XML files for each images which contains the annotation details of the that particular image.

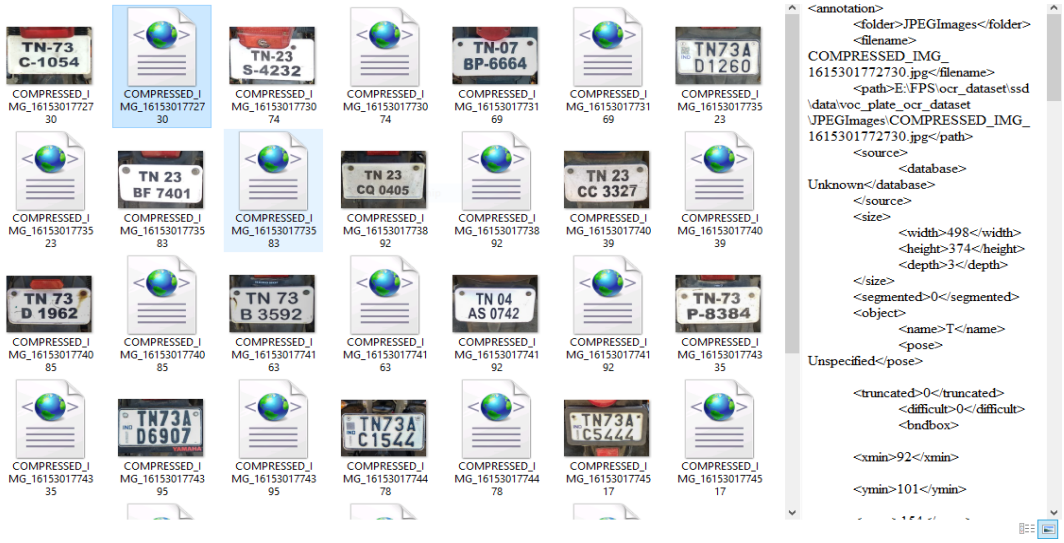


FIG5.4 annotations

5.2.4 NVIDIA Tensor RT

The core of NVIDIA® TensorRT™ is a C++ library that facilitates high-performance inference on NVIDIA graphics processing units (GPUs). It is designed to work in a complementary fashion with training frameworks such as TensorFlow, Caffe, PyTorch, MXNet, etc. It focuses specifically on running an already-trained network quickly and efficiently on a GPU for the purpose of generating a result (a process that is referred to in various places as scoring, detecting, regression, or inference). Some training frameworks such as TensorFlow have integrated TensorRT so that it can be used to accelerate inference within the framework. Alternatively, TensorRT can be used as a library within a user application. It includes parsers for importing existing models from Caffe, ONNX, or TensorFlow, and C++ and Python APIs for building models programmatically. There are many ways to train neural network for deep learning .We trained our neural network through

datasets(images of Licesence plate) in the Google Colabalatory which is converted into ONNX model.

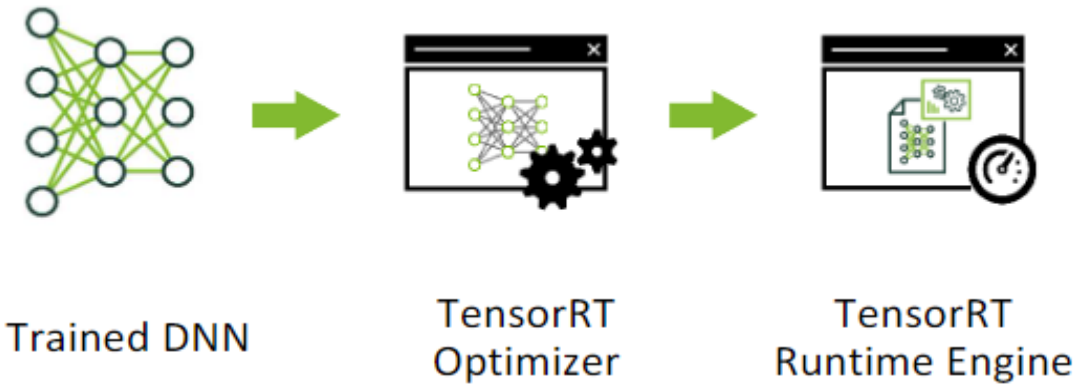


FIG 5.5. tensor RT

TensorRT is a high-performance neural network inference optimizer andruntimeenginefor production deployment.

5.2.2.1 How does TensorRT works

To optimize your model for inference, TensorRT takes your network definition, performs optimizations including platform-specific optimizations, and generates the inference engine. This process is referred to as the build phase. The build phase can take considerable time, especially when running on embedded platforms. Therefore, a typical application will build an engine once, and then serialize it as a plan file for later use.

5.2.5 GOOGLE COLAB

Google have released Colaboratory: a web IDE for python, to enable Machine Learning with storage on the cloud this internal tool had a pretty quiet public release in late 2017, and is set to make a huge difference in the world of machine learning, artificial intelligence and data science work.

FIG 5.6 google colab

5.2.6 ONNX Model

The Open Neural Network Exchange (ONNX) is an open format used to represent deep learning models. ONNX is supported by Amazon Web Services, Microsoft, Facebook, and several other partners. You can design, train, and deploy deep learning models with any framework you choose. The benefit of ONNX models is that they can be moved between frameworks with ease.

5.2.7 Pipeline

This project is developed based on the pipeline described below. From a set of data collected in practice to the problem you want to solve. For details in this project, we will use the dataset of Vietnamese license plates.

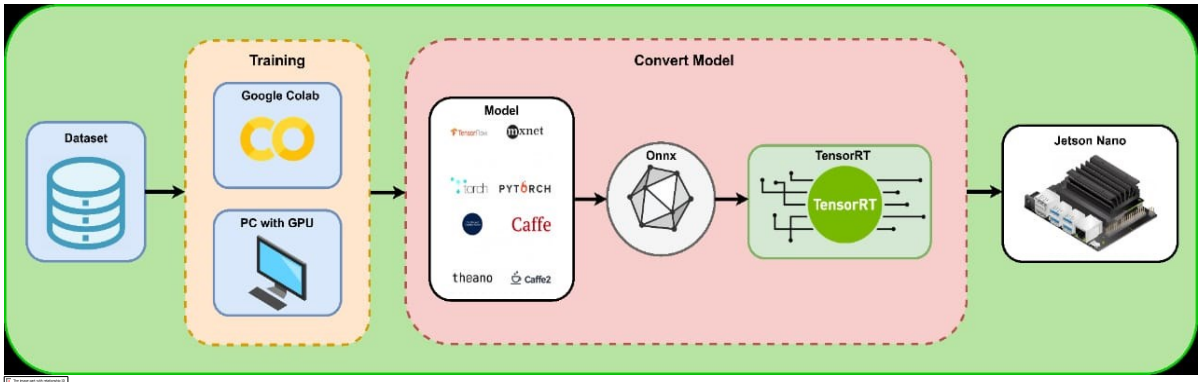


FIG 5.7 pipeline

First, you need to prepare a labeled dataset. Then train the object detection model with the GPU on Google Colab or your computer. Depending on the Deep learning Framework you use, it will output the model file in different formats. With ONNX you can convert most of the above formats to a single `.onnx` format. Then with TensorRT installed on the Jetpack Jetson Nano, you can run the object detection algorithms with high accuracy and FPS.

5.2.8 SSD MobileNet Architecture

The SSD architecture is a single convolution network that learns to predict bounding box locations and classify these locations in one pass. Hence, SSD can be trained end-to-end. The SSD network consists of base architecture (MobileNet in this case) followed by several convolution layers:

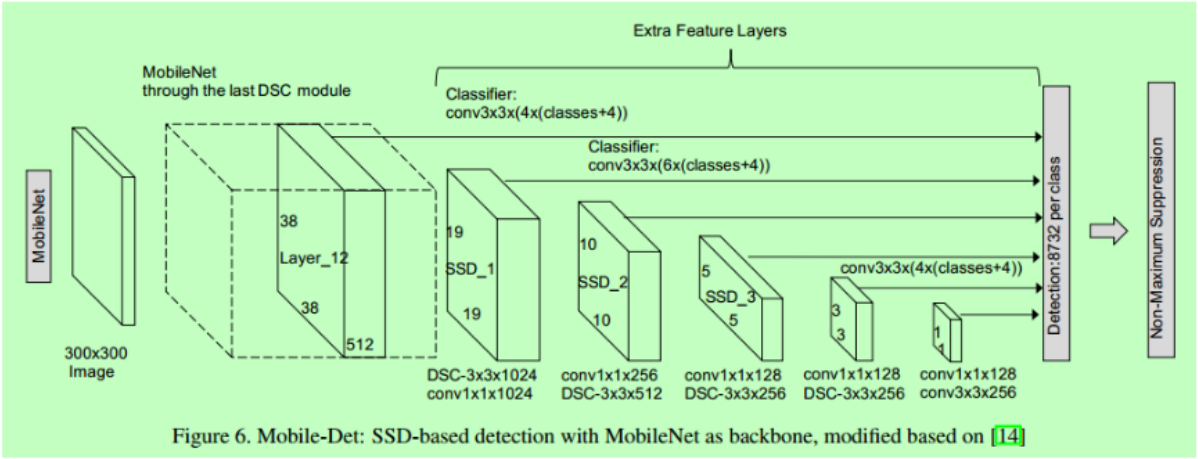


Figure 6. Mobile-Det: SSD-based detection with MobileNet as backbone, modified based on [14]

FIG 5.8 ssd

By using SSD, we only need to **take one single shot to detect multiple objects within the image**, while regional proposal network (RPN) based approaches such as R-CNN series that need two shots, one for generating region proposals, one for detecting the object of each proposal. Thus, SSD is much faster compared with two-shot RPN-based approaches.

5.2.9 Deploying a Deep Learning Network

This project uses NVIDIA TensorRT for efficiently deploying neural networks onto the embedded Jetson platform, improving performance and power efficiency using graph optimizations, kernel fusion, and FP16/INT8 precision.

Vision primitives, such as imageNet for image recognition, detectNet for object detection, and segNet for semantic segmentation, inherit from the shared tensorNet object. Examples are provided for streaming from live camera feed and processing images. See the API Reference section for detailed reference documentation of the C++ and Python libraries.

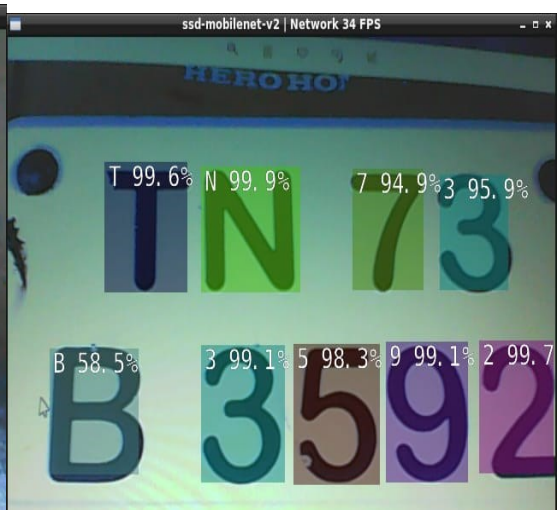


FIG 5.10 detection from ssd mobilenet-v

FIG 5.9 recognition from 2ssd mobilenet-v2

Fig 5.11 raw image from camera

fig 5.12 plate detection by ssd mobilenet-v2

Fig 5.13 cropped image by OCR

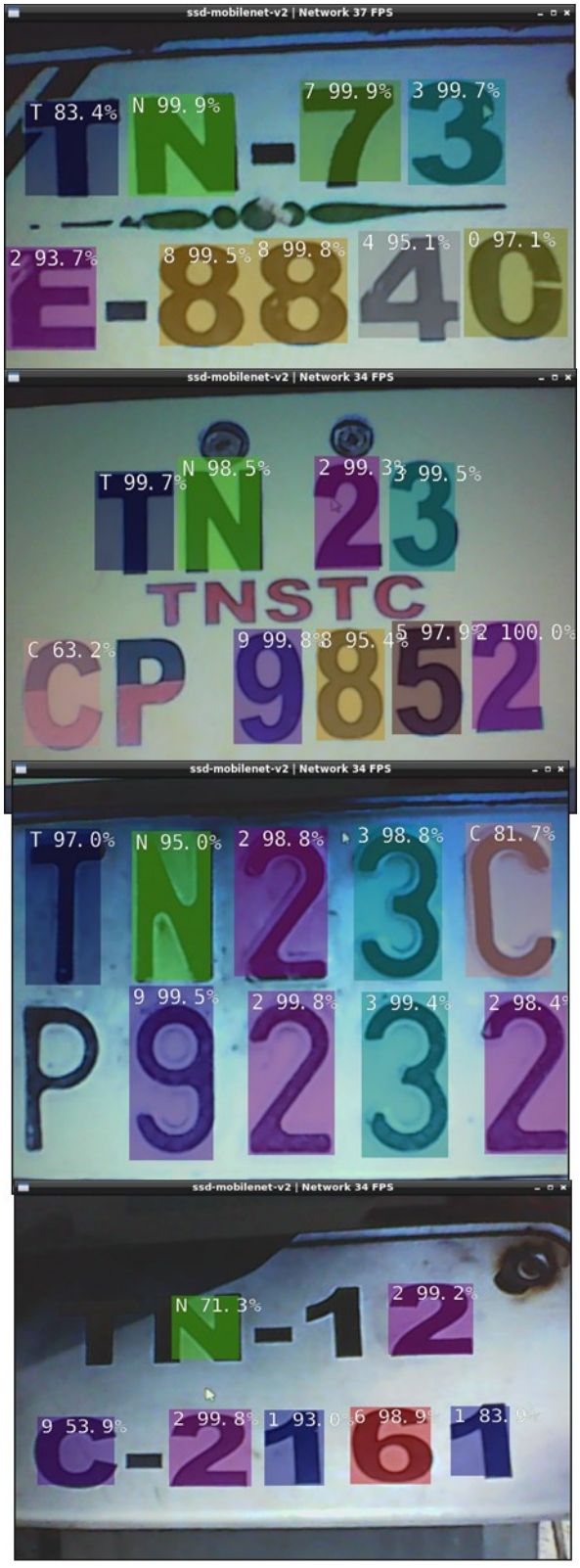


Fig 5.14 recognition from ssd mobilenet-v2

CHAPTER 6
TEST AND VALIDATION

6.1 GENERAL

This chapter deals with resultant process and its output. It also deals with how the live data are monitored and displayed. It describes about pros, cons and applications of the project.

6.2 RESULT OF THE PROJECT

Thus the detection and recognition of the live number plate are obtained, and in the results character in number plate where detected and recognition as characters text in number plate where obtained .In the end the data were saved in the Google spreadsheet which shows number,date,time.

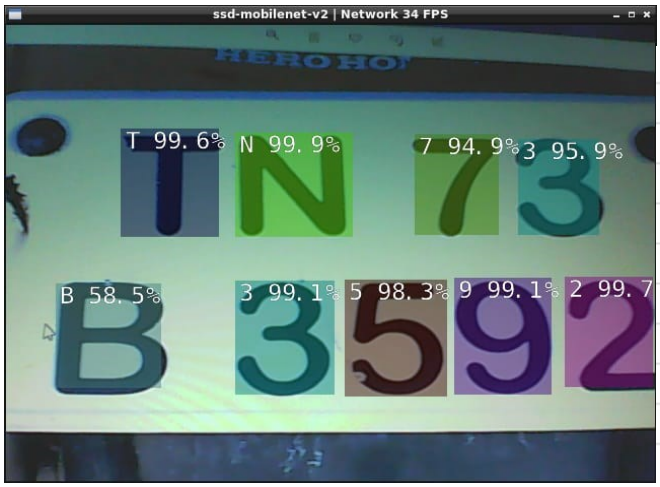


Fig 5.1 program output

plate	time	date
TN 73 B 3592	11.24.43	3/3/2021
TN 23 AB 4562	11.25.34	3/3/2021
TN 25 BB 6342	11.25.59	3/3/2021
TN 73 AA 7341	11.26.53	3/3/2021
TN 23 K 3742	11.27.34	3/3/2021
TN 23 B 5273	11.31.34	3/3/2021

Fig 5.2 Spreadsheet output

6.2.1 TESTING

Image Capturing Should be clear to train It should n't contains any shadows or any marks in it.

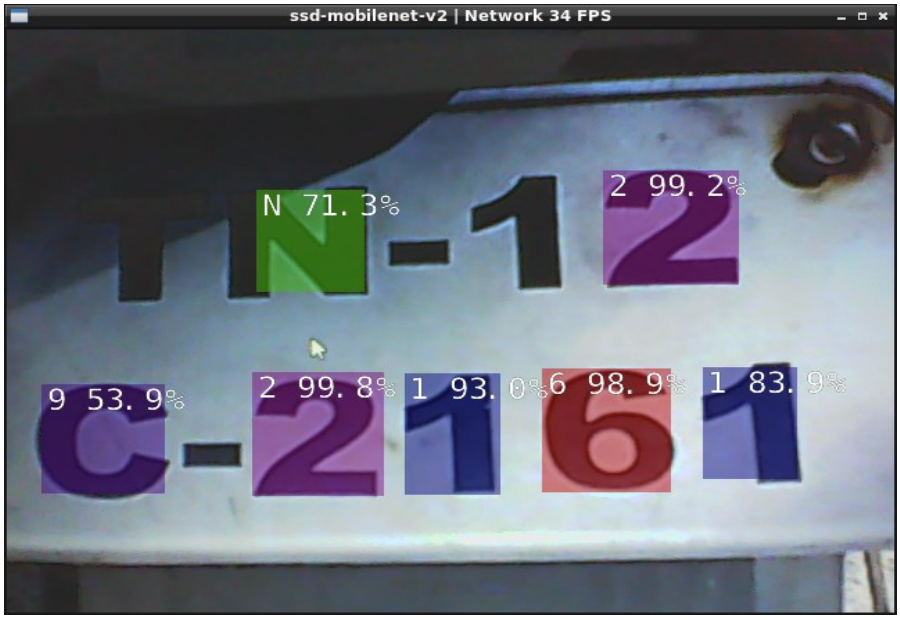


FIG 6.3 partially detected character of trained model

On Annotating process letters should be bounded properly if not then that letter doesn't detected like in below image.

FIG 6.4 partially detected character of trained model

There is no proper boundary for number plate in india. Many people follows various font types in number plate so it is difficult to implement in india

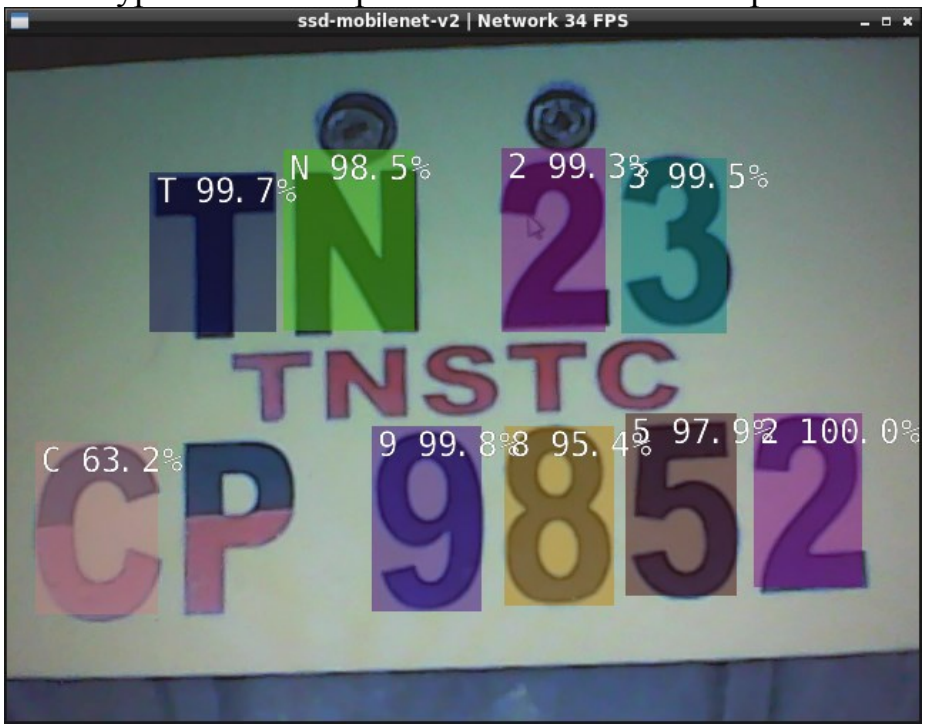


FIG 6.5 partially detected character of trained model

Characters in number plate should be clear. If it is damaged then the characters may be wrongly detected.

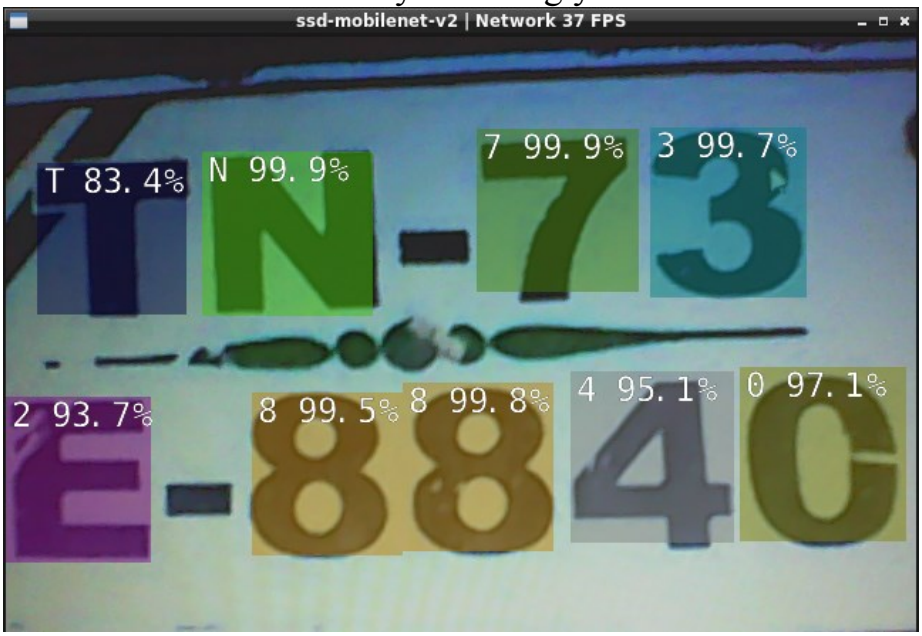


FIG 6.6 **partially detected character of trained model**

All Character in the number plate font size should be same.

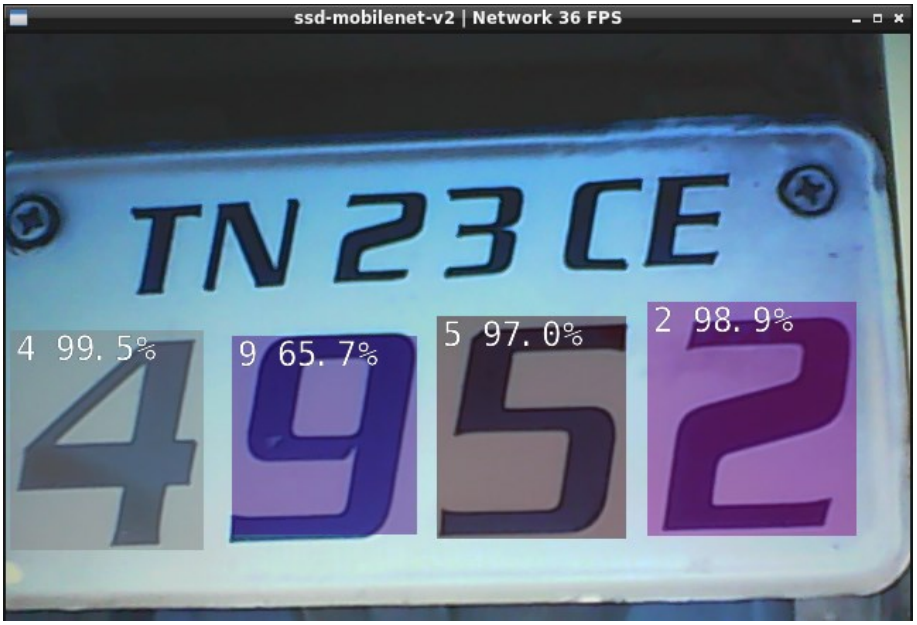


FIG 6.7 partially detected character of trained model

NO OF PLATE	**200**
NO OF PLATE DETECTED	**168**
NO OF PLATE UNDETECTED	**32**

6.3 PROS OF THE PROJECT

Surveillance and Monitoring

With the help of automatic number plate recognition software, you can easily avoid the stressful and time-consuming process of recording numbers manually. It is next to impossible to spot the right license plate number, especially when a car is speeding.

Better Security

Automatic license plate recognition is capable to deter constant traffic offences. It is helpful to reactive security like forensics, inspections, investigations, and legal matters.

Video footage and stills

A lot of automatic license plate detection systems offer both video footage and still images..

No or fewer needs for manned surveillance

With the reduced need for manned surveillance, some say it a con. But it is not true in all the ways.

CONS OF THE PROJECT

Privacy

It is true that records and images are stored and kept but it leads to some issues related to privacy. Usually, people are worried that the information of their whereabouts might be misused which are recorded in those footages. It can get into wrong hands or be subject to data thefts.

Extreme weather can affect the accuracy

Hindrances and extreme weather conditions can affect the accuracy of automatic license plate recognition software. Manned surveillance would be required because automatic security systems might not work.

APPLICATIONS

Banks and finance.

Public areas.

Transportation.

Mass retailing.

CHAPTER 7

CONCLUSION

Traffic control and vehicle owner identification has become major problem in every country. Sometimes it becomes difficult to identify vehicle owner who violates traffic rules and drives too fast.

Therefore, there is a need to develop Automatic Number Plate Recognition (ANPR) system as a one of the solutions to this problem.

This ANPR system is secure and accurate one. If this system is employed in our society then crimes can be minimized and records can be maintained. We can expect this system to create a larger impact in our environment.

FUTURE SCOPE

Today advances technology took Automatic Number Plate Recognition (ANPR) systems from hard to set up, limited expensive, fixed based applications to simple mobile ones in which “point to shoot” method can be used. This is possible because of the creation of software which ran on cheaper PC based and also non specialist hardware in which their no need to give pre-defined direction, angels, speed and size in which the plate would be passing the camera field of view. Also Smaller cameras which can read license plates at high speed, along with smaller, more durable processors that can fit in police vehicles, allowed law enforcement officers to patrol daily with the benefit of license plate recognition in real time.

CHAPTER 8
REFERENCE

[1] Hamey, L.G.C. and C. Priest, Automatic Number Plate Recognition for Australian Conditions. Proceedings of the Digital Imaging Computing: Techniques and Applications, 2005: p. 8.

[2] Chen, Z.-X., et al., Automatic License-Plate Location and Recognition Based on Feature Salience. IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, 2009. 58(7): p. 3781-3785.

[3] Qadri, M.T. and M. Asif, Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition. International Conference on Education Technology and Computer, 2009: p. 335 - 338.

[4] Meier, R., Professional Android 2 Application Development2010, Indianapolis, Indiana: Wiley Publishing, Inc. 399.

[5] Maarif, H.A.-Q. and S. Sardy, Plate Number Recognition by Using Artificial Neural Network. Prosiding Semiloka Teknologi Simulasi dan Komputasi serta Aplikasi, 2006: p. 176-182.

[6] Badr, A., et al., Automatic Number Plate Recognition System. Annals of the University of Craiova, Mathematics and Computer Science Series, 2011. 38(1): p. 62-71

[7] Kim, K.K., et al. Learning-based approach for license plate recognition. in Neural Networks for Signal Processing X, 2000. Proceedings of the 2000 IEEE Signal Processing Society Workshop. 2000.

[8] Dai, Y., et al. A high performance license plate recognition system based on the web technique. in Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE. 2001.

[9] Khalifa, O., et al., Malaysian Vehicle License Plate Recognition. The International Arab Journal of Information Technology, 2007. 4(4): p. 359-364.

[10] Duan, T.D., et al., Building an Automatic Vehicle License-Plate Recognition System. Intl. Conf. in Computer Science – RIVF'05,

Vietnam, 2005: p. 59-63.
[11] Yo-Ping, H., L. Shi-Yong, and C. Wei-Po. A template-based model for license plate recognition. in Networking, Sensing and Control, 2004 IEEE International Conference on. 2004.
[12] Hegt, H.A., R.J. de la Haye, and N.A. Khan. A high performance license plate recognition system. in Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on. 1998.
[13] Xiaojun, Z., F. Bensaali, and R. Sotudeh. OCR-based neural network for ANPR. in Imaging Systems and Techniques (IST), 2012 IEEE International Conference on. 2012.
[14] Radzi, S.A. and M. Khalil-Hani, Character Recognition of License Plate Number Using Convolutional Neural Network. International Visual Informatics Conference, IVIC, 2011. 2: p. 45-55.
[15] Ying, W., et al., An Algorithm for License Plate Recognition Applied to Intelligent Transportation System. Intelligent Transportation Systems, IEEE Transactions on, 2011. 12(3): p. 830-845.
[16] Ozbay, S. and E. Ercelebi, Automatic Vehicle Identification by Plate Recognition. World Academy of Science, Engineering and Technology, 2007. 9: p. 778-781.
[17] Kasaei, S.H., S.M. Kasaei, and S.A. Kasaei, New Morphology-Based Method for Robust Iranian Car Plate Detection and Recognition. International Journal of Computer Theory and Engineering 2010. 2(2):
p. 264-268.
[18] Smith, R. An Overview of the Tesseract OCR Engine. in Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on. 2007.
[19] Brunelli, R., Template Matching Techniques in Computer Vision: Theory and Practice 2009, United Kingdom: Wiley