UNIVERSITY OF SIALKOT

A CHARTERED UNIVERSITY

Faculty of Computing & IT

University of Sialkot

Automatic Number Plate Recognition (ANPR) Based Toll Collection System

Session: BS – Information Technology

Fall 2019-2023

Supervisor: Dr. Adven Gill

Submitted By

UmairAltaf	*19101003-093*
Haseeb Ali	*19101003-097*
UmairYaqoob	*19101003-098*
Muhammad Waseem	*19101003-162*

Faculty of Computing & IT
University of Sialkot

STATEMENT OF SUBMISSION

This is certify that________________RollNo.__________________and ________________Roll No__________________ and __________________ Roll No.__________________________and______________________Roll No. _________________have successfully completed the final year project named as- **Automatic Number Plate Recognition (ANPR) Based Toll Collection System**

at the Department of Information Technology, University of Sialkot, to fulfill the requirement of the degree of Bachelor in Information Technology.

____
ProjectSupervisor

____
ProjectManagementOffice
FacultyofC&IT-USKT

____
HeadoftheDepartment

Acknowledgement

We truly acknowledge the cooperation and help make by___________________ Head, Department of Information Technology, University of Sialkot. He has been a constant source of guidance throughout the course of this project. We would also like to thank Mr.______________________for his help and guidance throughout this project. We are also thankful to our friends and families whose silent support led us to complete our project.

1-Umair Altaf

2-Umair Yaqoob

3-Haseeb Ali

4-Muhammad Waseem

Date:May 9, 2023

Table of contents

Abstract

ANPR (Automatic Number Plate Recognition) is a procedure in machine learning technology that captures and identifies an image of a number plate of a vehicle automatically.Traditionally this process is difficult as it is hard to manage a high number of vehicles, time consuming, costly and takes a lot of time in verification.There are many applications of this system that are being used in the world.The aim of this research is to develop a system which uses ANPR technology with a custom dataset for training, OCR (optical character recognition) and image processing using CCTV cameras that are installed onsite. The detected number plate is processed and character extraction is done and then those characters are sent to the database to see if the record exists if it is not existent it is added to the details with the time of vehicle entry to calculate toll in parking lots and generate a receipt. As user had to pay full amount previously , after implementing this system a fair charge of toll will be calculated based on the time spent in the parking lot .Another highlight of this research is to develop an algorithm that is trained on the Pakistani Format of number plates as it has not been implemented in this field. After this study the goal is to develop a lightweight system that can be implemented on a smaller scale, cheap and easy to understand by everyday user and uses less computational power

CHAPTER 1: INTRODUCTION

1.1. INTRODUCTION

The number of vehicles are increasing due to the continuous increase in the population, the number of vehicles is also increasing, and due to the increase in the number of vehicles, their security and parking problems are also faced. In every country, the identification of the vehicle is checked from the number plate of the vehicle, but it is quite difficult to store the record of the number plate manually in the parking lots. As seen in [5], number plate racking has been a very interesting and challenging research topic all over the world.

The two main objectives are to create a system that automatically detects the number plate of vehicles entering the parking lot and calculates the time of entry and exit of the vehicle and charges the parking fee accordingly.

In the existing system, one person had to note down the number of the vehicles manually, which was quite laborious work.It was done with a pencil or paper or in a computer database, the records had to be noted down by hand. This system was also a cause of dissatisfaction for the vehicle owner to some extent, the user suspected that charges were unfair

And on the other hand, if someone used the parking only for a few minutes, i.e. a car parked for a few minutes, had to pay the same amount as the person who used the parking for eight to ten hours.

- In the old system, manually vehicle number plate records were stored in a database
- Whether a user used the parking for five minutes or ten hours, both had to pay the same amount
- Vehicles that did not have number plates were quite difficult to park

1.2. PROBLEM STATEMENT

In [1],[2],vehicle records were stored based on their number plates. And vehicles without number plates used to face a lot of difficulties. This system will also provide facilities to vehicles without number plates. And on the other hand, if someone used the parking only for a few minutes, i.e. parked his car in the parking only for a few minutes, then he had to pay the same amount as the person who used the parking for eight to ten hours. According to the system, as much money will be charged as long as the vehicle is parked in the parking lot.Another gap that was found was that most systems were targeted towards specific countries and we only found a few that targeted Pakistani number plates.

1.3. HISTORICAL BACKGROUND

In the old system, all the work was done manually, that is, the number of the vehicles was noted with the help of a paper and a pencil. After that, as soon as the system was slightly updated, instead of storing vehicle numbers with the help of pencil and paper, it started to be stored in a database, and that too was done completely manually. Number plate recognition systems have also been implemented in some places, but there is only a limited number of use cases.

1.4. OBJECTIVES

1. To create a system that automatically detect the number plate of vehicles entering the parking lot and calculates the time of entry and exit of the vehicle and charges the parking fee accordingly.The interval between entry and exit will be calculated and the fee will be charged based on this time.

2. To make a system will also provide a lot of convenience to those vehicles that do not have a number plate on them.

1.5. PROJECT SCOPE

This system is designed to recognize Pakistan format number plates, by training the algorithm on our custom datasets. the platform can mainly recognize Pakistan format number plates only. If the number plate is in any other format which does not follow the Pakistani standard, then the user will be treated as a vehicle without a number plate.

1.6. RELATED WORK

In [3], the author proposed that an integrated electronic toll collection system using raspberry pi and IoT was used to connect entire system with office database in real time. The integration of the system was enhanced by elaborating existing system with IoT for transmitting data received from Pi camera and then using image processing using python ,extracting the information.

In[4], Authors presented an ANPR framework which used OCR Tesseract to address the vast heterogeneity and assortment of number plate genres across the provinces in Pakistan.the proposed algorithm was tested on low-resolution images which were collected by considering the variations in the environment illumination and image orientation.the proposed preprocessing techniques were applied to the localized plate image, and then the processed image was passed to the OCR Tesseract for recognition The obtained score for the plate extraction using YOLOv3 was 94.3% and YOLOv4 was 99.5% on 0.50 thresholds.The introduced pipeline with the YOLOv4 model would take a 0.80-second average computation time per image and gave an average accuracy score of 73%

1.7. SUMMARY

The goal is to create a system that will be beneficial for the vehicle owner and the car parking manager.An Automatic number plate recognition and toll collection system that will recognize the Pakistani number plate and store it in the database and calculate the fee accordingly, the time the vehicle spends in parking that means the fee will be charged by calculating both the time the vehicle entered the parking lot and the time the vehicle exited the parking lot.

CHAPTER 2: LITRERATURE REVIEW AND SYSTEM SPECIFICATIONS

2.1. LITERATURE REVIEW

In [1], the author proposed that the system used OCR and processed the captured image into grayscale. Then, morphological operations were performed. Optical character recognition (OCR) was used for character recognizing using template matching. After recognition of data, the owner would receive a notification on mobile through GSM system. A Webcam was used to capture the number plate of incoming vehicles. Verification was done and deduction of toll tax was carried out hence allowing the vehicles to pass.

In [2], the author elaborates that the system worked by using Optical Character Recognition(OCR) which had a more positive rate in the result and in this model CNN was used for training the data for good accuracy in the result.The OCR consisted of six steps that would be done for extracting the text from the number plate such as Image Acquisition, Pre-Processing, Segmentation, Feature Extraction, Training a Neural Network and Post-Processing. The text extracted from the number plate was then matched with the database and it sent the OTP to the user for identifying the theft.

In [3], the author proposed that an integrated electronic toll collection system using raspberry pi and IoT was used to connect entire system with office database in real time. The integration of the system was enhanced by elaborating existing system with IoT for transmitting data received from Pi camera and then using image processing using python ,extracting the information.

In [4], author proposed an automatic CNN architecture design method by using genetic algorithms, to effectively address the image classification tasks. The point to note of the proposed algorithm was its “automatic” characteristic. The proposed algorithm was validated on widely used benchmark image classification datasets, by comparing to the state-of-the-art peer competitors covering eight manually-designed CNNs, seven automatic & manually tuning and five automatic CNN architecture design algorithms. The experimental results indicated the proposed algorithm outperformed the existing automatic CNN architecture design algorithms in terms of classification accuracy, parameter numbers and consumed computational resources. The proposed algorithm also shows the very comparable classification accuracy to the best one from manually-designed and automatic & manually tuning CNNs, while consumes much less of computational resource.

In [5], authors investigated the performance of two state-of-the art CNN algorithms, namely Faster R-CNN and YOLOv3, in the context of car detection from aerial images. These algorithms trained and tested these two models on a large car dataset taken from UAVs. It is demonstrated in this paper that YOLOv3 outperforms Faster R-CNN in sensitivity and processing time, although they are comparable in the precision metric.although both algorithms scored 99% accuracy the yolo v3 outperforms faster R-CNN in terms of sensitivity.

In [6], an effective nighttime vehicle detection approach was designed.The original night images were captured first and enhanced by an optimal MSR algorithm.Then a pretrained YOLO v3 network is selected an fine tuned. Finally the detection network was used to detect vehicles from nighttime images.The proposed method outperformed Faster R-CNN and SSD. The average precision of proposed model reached 93.66% which is 6.14% and 3.21% higher than Faster R-CNN and SSD.

In [7], a dilated CNN model which is built through replacing the convolution kernels of traditional CNN by the dilated convolution kernel is proposed.The dilated CNN model is tested on the Mnist handwritten digital recognition data set. To solve the detail loss problem in the dilated CNN model, the hybrid dilated CNN (HDC) is built by stacking dilated convolution kernels with different dilation rates, and then the HDC model is tested on the wide-band remote sensing image data set of earth's terrain. The results show that under the same environment, compared with the traditional CNN model, the dilated CNN model reduces the training time by 12.99% and improves the training accuracy by 2.86% averagely, compared with the dilated CNN model, the HDC model reduces the training time by 2.02% and improves the training and testing accuracy by 14.15% and 15.35% averagely. Therefore, the dilated

CNN and HDC model proposed in this paper can significantly improve the image classification performance.

In [8], CNN models are built to evaluate performance on image recognition and detection datasets. The algorithm is implemented on MNIST and CIFAR-10 dataset and its performance is evaluated. The accuracy of models on MNIST is 99.6 %, CIFAR-10 is using real-time data augmentation and dropout on CPU unit.

In [9], a great approach to detection of fruits using deep Convolutional neural networks was introduced. The aim was to build an accurate, fast and reliable fruit detection system using machine learning facts. The proposed system has applied Convolutional neural network (CNN) to the tasks of detecting fruit images. A dataset was constructed that contained most frequent fruit items in a publicly available fruit-logging system, and used it to evaluate detection performance. CNN showed higher accuracy than did traditional support-vector-machine-based methods with handcrafted features. For fruit image detection, CNN also showed significantly higher accuracy than a conventional method did. All the data in the data set were collected using Smart-phone camera and believed to be unique in every sense. The study had shown an accuracy of 99.89% which turned out to be promising.

In [10], author proposed an HSI classification method based on the 2D–3D CNN and multibranch feature fusion. First 2-D CNN and 3-D CNN were combined to extract image features. Then, by means of the multi branch neural network, three kinds of features from shallow to deep were extracted and fused in the spectral dimension. Finally, the fused features werer passed into several fully connected layers and a soft max layer to obtain the classification results. In addition, our network model utilizes the state-of-the-art activation function Mish to further improve the classification performance. Experimental results, conducted on four widely used HSI datasets, indicated that the proposed method achieved better performance than the existing alternatives.

In [11], the author proposed a system for image classification using SVM and CNN. First a small scale dataset was used for SVM and accuracy of 93% was achieved.Then the data was augmented and the accuracy decreased to 83%. The author also implemented CNN for image classification and gained a score of 93.57% on the same dataset.The conclusion was reached that CNN is better than traditional approaches .5 classes were used.

In [12], the study proposes improving the vehicle classification accuracy of Yolo, and developing a novel bounding box (Bbox)-based vehicle tracking algorithm. For this purpose, a new vehicle dataset was prepared by annotating 7216 images with 123831 object patterns collected from highway videos. Nine machine learning-based classifiers and a CNN-based classifier were selected. Next, the classifiers were trained via the dataset. One out of ten classifiers with the highest accuracy were selected to combine to Yolo. This way, the classification accuracy of the Yolo-based vehicle detector was increased from 57% to 95.45%. The vehicle counting results show that the vehicle counting accuracy of the developed approach was improved by 13.25% .

In [13], A novel approach using light Convolutional neural network architecture for recognizing defects in EL images was presented which achieved state of the art results of 93.02% on solar cell dataset of EL images. It required less computational power and time. It can work on an ordinary CPU computer while maintaining real time speed. It took only 8.07 ms for predicting one image. Data augmentation operations were used to deal with data scarcity.An average accuracy of 93% was achieved. The proposed framework was experimentally applied in lab and can help for automatic defect detection in field and industry.

In [14], the functionality of pre-trained CNN models utilized as feature-extractors followed by different classifiers for the classification of abnormal and normal chest X-Rays is appraised. The authors analytically determine the optimal CNN model for the purpose. Statistical results obtained demonstrated that pre-trained CNN models employed along with supervised classifier algorithms can be very beneficial in analyzing chest X-ray images, specifically to detect Pneumonia.

In [15], The authors proposed a technique to improve the newly released, YOLOv4 detector, specifically, for vehicle tracking applications using some existing methods such as optimising anchor box predictions by using k-means clustering. . The model was also compared with the existing models on a number of performance metrics such as - precision, recall, F1 score, mean average precision, and the average IoU. The experimental results show that the SOTA model which already has real-time object detection capabilities can be further improved for highly targeted use cases. Final combined model gave benchmark results with an mAP of 67.7% (10%-point higher than base model) on the DETRAC-test dataset.

In [16], a robust method to detect vehicles in video frames based on CNNs is proposed which provides an almost real-time performance and impressive accuracy. To overcome the challenges of building a precise vehicle detection model from still images. The main architecture of a pre-trained ResNet-50 residual network to Faster Region-based Convolutional Neural Network (Faster R-CNN) was transformed in this work. Experimental results showed that the system's sensitivity factor is 0.985 and it needed an average of 74 milliseconds to detect vehicles in real-condition data. Consequently, the proposed method can provide acceptable results on real-condition data in terms of accuracy and execution time.

In [17],the authors proposed a ANPR system, during the implementation of the ANPR system,hardware and software specifications were both considered. From the point of view of mobility and ease of deployment hardware, smartphones had the upper hand.
The proposed ANPR system was designed and implemented to Android-based smartphone the whole system used OCR to determine license plate information from the input image. The software also acted as a database for various users, their media and related information, including Parking and vehicle-related violations in time
transit. This system also provided the basis for the law enforcement officer to issue and resolve challans for offenders. There was also a feature to control toll collection agency.The proposed system used ANN(artificial neural networks) and KNN(K-nearest neighbour) Algorithm and achieved an overall score of 73.06%.

In [18],the authors used open computer vision techniques and effective OCR with python software. license plate character recognition language. Pattern matching was used to improve license plate accuracy.It was also observed that several factors play an important role in OCR technique-dependent pattern matching, such as Noise in image and camera distance, font type and background, etc .This technique was used for alphanumeric characters by digital image processing and OCR.

In [19], the authors made a demo based on a variety of license plates of other nations. Yolov4, using the CNN structure, was used to prepare the image data set. After different image pre-processing procedural and morphological changes, character recognition was performed using Tesseract OCR. About license plate discovery, the recommended framework covered an accuracy rate of 92% and 81% in character recognition. Which suggested that a higher accuracy was achieved in hindsight of existing models.

Research gap

In the literature review many novel algorithms have been used to detect number plates of cars which offered good accuracy e.g 80-95%.In [6] RFID is used to make a system that runs 24 hours and helps in toll collection.in [11][19][18] OCR is used for number plate recognition.It has been observed in the literature review that many of the existing systems are hard to develop and require a large amount of computational power and are very costly to setup as they are designed for use on a larger scale.

Automatic toll collection is an area that has been targeted less in previous researches so our aim is to develop a light weight automatic number plate recognition system that automates the toll collection process by automatically generating a receipt which has details of entry and exit and deducted charges according to time spent which can be used on a small scale for example in the parking lots of hospitals and universities.

Another gap that was found was that most systems were targeted towards specific countries and we only found a few that targeted Pakistani number plates.As we use a different format of number plate, it is hard to use a existing datasets which are successful in identifying Pakistani number plates.

List of literature studies shown in **Table 1.**

Paper name	Objectives	Discussion/ Remarks	Accuracy	Year	Proposed method
Toll Tax Collecting System Using Optical Character Recognition [1]	to decrease the traffic by reducing the manual work and also time saving as there will be no giving and taking of money or the receipt	Selected model may not recognize text over a line or text in a block	Not mentioned	2020	OCR, morphological operations, noise removal, text enhancement, binirization
Number Plate Recognition Using CNN For Identification of Theft in Toll Collection System [2]	To make an efficient license plate recognition system for use in the toll booth for identifying theft of vehicles	unclear image would wipe out some important information causing the incorrect crop of the license plate	99%-character recognition 95% OCR 87.5% plate region extraction	2021	Binarization, Conditional Random Field, Optimal Character Recognition, OpenCV
IOT and Machine Learning based Automatic Toll Collection System [3]	The objective was to design an efficient Vehicle Number Recognition System & to implement it for automatic toll tax collection.	Wi-Fi or internet service availability was not ensured making the system only work with registered vehicles	Not mentioned	2021	Plate detection, image processing, authentication, Database cross check
Automatically Designing CNN Architectures Using Genetic Algorithm for Image Classification [4]	To make it easier for new users to design CNN architecture according to their own needs	Fitness evaluation can be improved	Not mentioned	2020	CNN, Variable length encoding, CIFAR 10, CIFAR 100
Car Detection using Unmanned Aerial Vehicles: Comparison between Faster R-CNN and YOLOv3 [5]	To compare two algorithms and find which performs better in image detection	Dataset can be improved, and it has capability to include more parameters	Precision (tpr) 99.73% in YOLO V3 99.66% in faster R-CNN	2018	YOLO V3, R-CNN
A Nighttime Vehicle Detection Method Based on YOLO v3 [6]	To enhance the accuracy of nighttime image detection	Suitable resources are discussed	MSR 93.66%	2020	MSR, YOLO V3, Faster R-CNN, SSD

A Dilated CNN Model for Image Classification [7]	Comparing HDC model with dilated CNN for image detection	Suitable resources are discussed	98.35% on 100o rounds 98.78% on 5000 rounds	2019	CNN, dilated CNN, HDC model
Convolutional Neural Network (CNN) for Image Detection and Recognition [8]	To use CNN models for image detection problems	The accuracy of training can be improved by adding more hidden layers	80.17% on CIFAR-10 after 50 epochs 99.6% on MNIST	2018	MNSIT, CIFAR 10 Augmentation
Convolutional Neural Networks (CNN) for Detecting Fruit Information Using Machine Learning Techniques [9]	To use CNN for fruit detection	The image dataset can be increased, and more fruit classes should be added	99.89% on 30 epochs	2020	CNN, ADAM optimizer
Hyperspectral Image Classification Method Based on 2D–3D CNN and Multibranch Feature Fusion [10]	to utilize the interband correlations of HSIs which is not possible in traditional 2-D CNN	Fully connected layers had too much training parameters	Mish activation function increased the accuracy by 1.8169%	2020	2D-3D CNN, Mish activation function
Image Classification using SVM and CNN [11]	SVM (traditional) comparison with CNN (convolutional neural network).	Dataset and the number of classes can be increased	83% with SVM 93.57% with CNN	2020	SVM, CNN
A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways [12]	To improve the accuracy of the existing algorithm.	Suitable resources are discussed	95.45% with YOLO	2021	YOLO, best classifier, CNN
CNN based automatic detection of photovoltaic cell defects in	To make a lightweight architecture for detection defects in photovoltaic cells	Suitable resources are discussed	93.025% on 64 batch size	2019	CNN, TensorFlow, open CV,

electroluminescence images [13]					
Pneumonia Detection Using CNN based Feature Extraction [14]	To improve medical adeptness where radiologists are not available at all times	Model uses a lot of convolutional layers, so computation is increased	80% accuracy with SVM (rbf kernel) Classifier	2019	CNN, K-nearest Neighbour, SVM
REFINING YOLOV4 FOR VEHICLE DETECTION [15]	To improve the newly released YOLOv4 detector,	Suitable resources are discussed	mAP of 67.7%	2020	YOLO v4, SAM, SAT, ABK
A Robust Vehicle Detection Approach based on Faster R-CNN Algorithm [16]	Developing a Real time vehicle detection algorithm using CNN to increase accuracy	Suitable resources are discussed	98.5% accuracy rate in correct classification of vehicles in real-time	2019	DNN, CNN, Res-Net 50, Faster R-CNN
Automatic Number Plate Recognition System Using Super-Resolution Technique [17]	The objective was to make a portable ANPR system that used an android phone using Artificial neural network	Accuracy could be increased	An average score of 73.02% was achieved.	2021	ANPR (automatic number plate recognition), MySQL (optical character recognition), ANN (Artificial neural networks), KNN (k-nearest neighbor) algorithm
An Efficient Approach for Automatic Number Plate Identification Using Optical Car Recognition (OCR) Technique [18]	The focus of this paper was to propose an algorithm which is more effective technique for vehicle number plate identification based on open computer vision (CV) and OCR techniques which give better performance in	The techniques used in this system were sensitive to different types of objects for identification	Not mentioned	2020	OCR (optical character recognition), CV (computer vision)
Number Plate Detection Using YOLOV4 and Tesseract OCR [19]	Objective was to use the YOLO v4 algorithm in the ANPR system to improve accuracy	Suitable resources are discussed	98% approval rate with a blunder rate of less than 1.	2022	ANPR (automatic number plate recognition), YOLO v4 (you only look once) algorithm version 4, Tesseract OCR (optical character recognition)

2.2. TOOLS

2.2.1. HARDWARE

- Webcam
- Windows Pc
- 16 gb ram
- Intel core or AMD X64 processor

2.2.2. SOFTWARE

- Anaconda Python distribution
- Open CV libraries
- Python Tesseract libraries
- PyCharm community python IDE
- Windows 8, 8.1 ,10
- Firebase
- Google Chrome
- Cascade trainer GUI
- Tesseract OCR

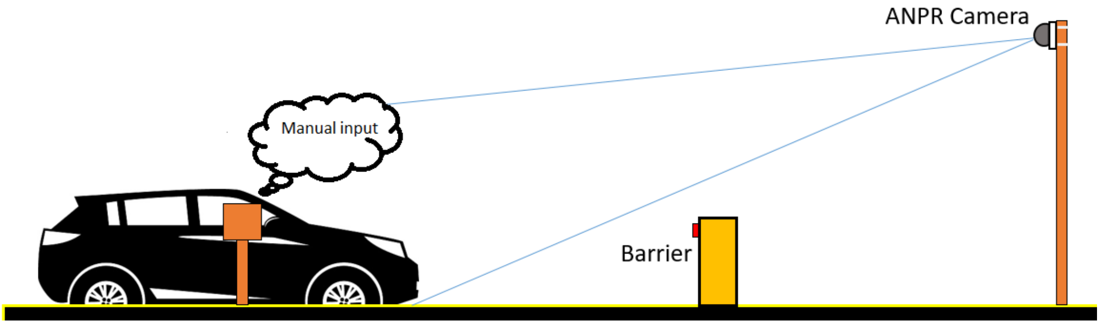
2.2.3. SIMULATION

- Python Virtual environment
- Google colab workspace

CHAPTER 3: DESIGN AND METHODOLOGY

3.1. SYSTEM MODEL

Figure 3.1.1. shows basic system model



System consists of a camera installed on site and a input device(id/face recognition) in case of manual entry. When image is detected and record is saved user can go into the lot and when exiting same process is repeated

3.2. WORKING OF PROJECT

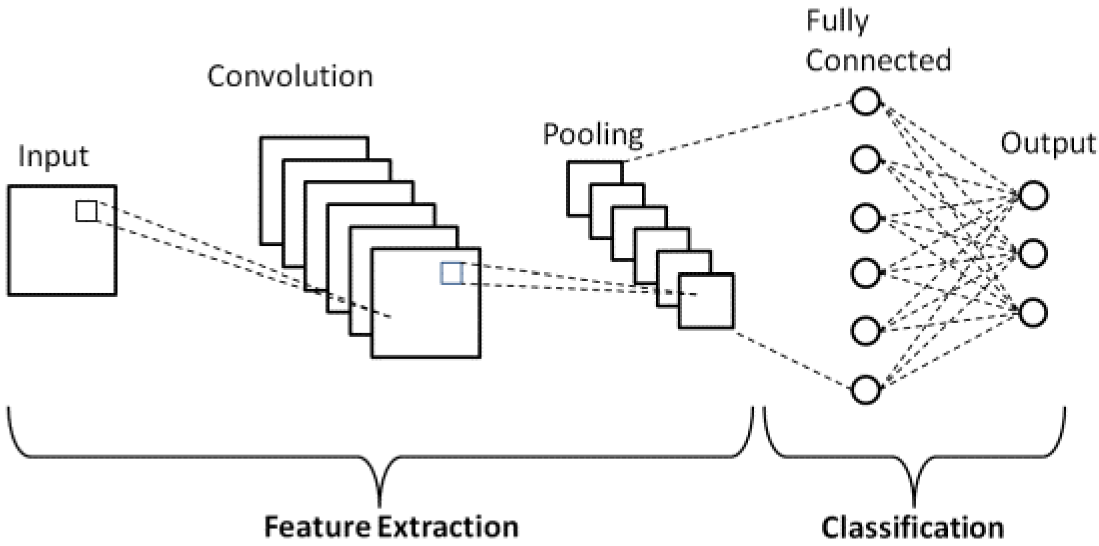
The proposed model takes input from a installed webcam or camera in the parking lot and then identifies the number plate on the vehicle.The captured picture is then per-processed to extract text.The text is then sent to the database and saved.The time is saved and when the vehicle is exiting and the number plate record is checked from the database and if the record exists , toll amount is calculated and a receipt is printed,In case of manual entry ID card number of the person is saved and when the vehicle is exiting the toll is calculated and receipt is printed

3.3. METHODOLOGY

TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training of deep neural networks. Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library. Both of these libraries provide API's used for building and training models. The Keras library is used in this project to build a CNN.

CNN or Convolutional neural network is a class of Artificial neural networks (ANN) which work like a brain and are composed of connected nodes called artificial neurons. CNN learns directly from data. A CNN can have multiple layers, each of which learns to detect the different features of an input image. A filter or kernel is applied to each image to produce an output that gets progressively better and more detailed after each layer.

Figure 3.3.1. Shows working of CNN

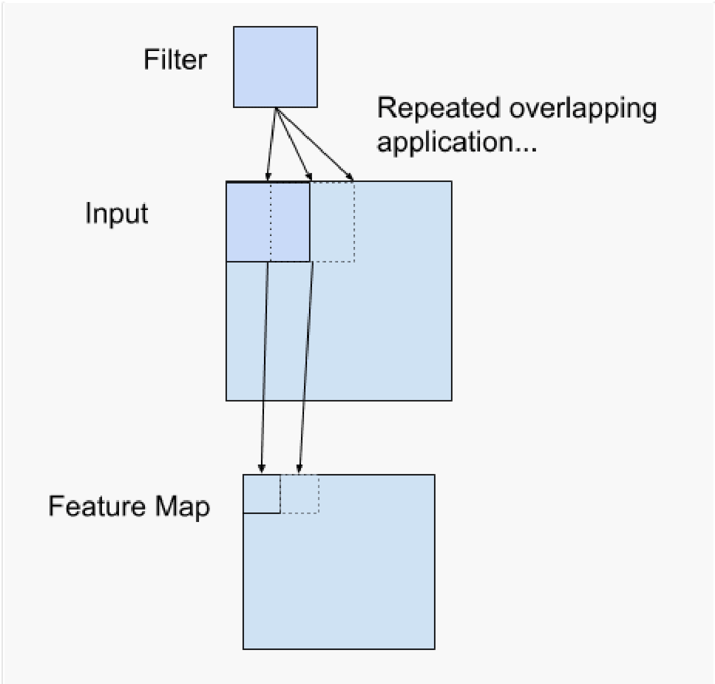


A CNN contains three layers

1) Convolutional layer

A Convolutional layer is like a filter that takes in an image and looks for specific features. It breaks the image down into smaller parts and checks to see if certain patterns exist. These patterns are learned through training. When a filter convolves with an image, it produces an activation map that shows where the pattern was found. This process is repeated with multiple filters to create a set of activation maps that are combined to produce the output of the layer.

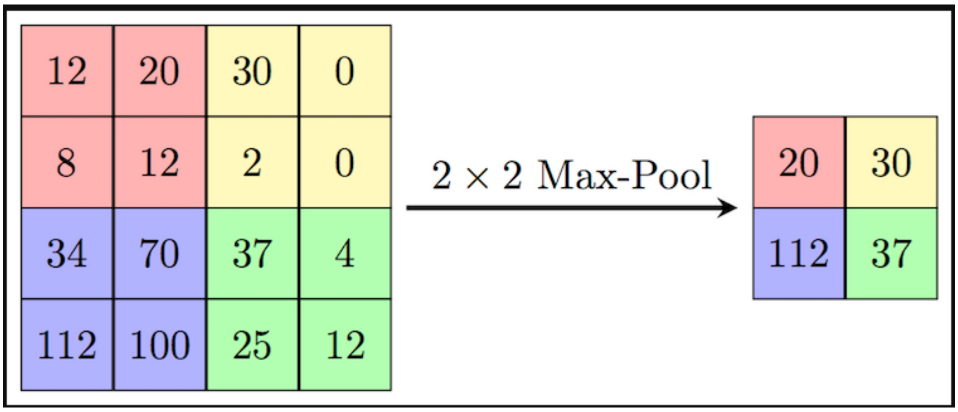
Figure 3.3.2. Shows working of Convolution layer



2) Pooling layer

The pooling layer also sweeps a filter across the input image, it reduces the number of feature map which also results in some information loss. But on a positive side, this layer reduces complexity and improves the efficiency of the CNN.

Figure 3.3.3. Shows working of pooling



3) Fully-connected layer

The input to the fully connected layer is the output from the final Pooling or Convolutional Layer, which is flattened and then fed into the fully connected layer.

In order to make a CNN model we need data, in the case of this project a custom dataset is made for object identification and detection. The working is as follows

1) Data Collection
2) Pre-processing
3) Data Augmentation
4) Uploading data to google drive
5) Getting rid of unwanted data
6) Train, Test, Validation split
7) Building the model
8) Testing the model
9) Saving the model for future use
10) Image acquisition
11) Number plate area identification and extraction
12) Text extraction
13) Database operations
14) Android app Development

The purpose of this project is to build a system that recognizes and saves data of the number plates in real time and then generates a toll receipt.This system can be used in parking lots and it can run on a PC and with external and integrated camera.Once the number plate is detected its characters are extracted and sent to database to compare.If the record exists it means that vehicle is exiting and if a new record is added it means that a vehicle has just entered the parking lot.

Properties:

Data Collection:

The first step involves collecting data. Data is collected in the form of images. In this case 233 images of cars with Pakistani number plate were downloaded from the internet and 360 images were captured from mobile camera for a total of 593 images.

Pre-processing:

After collecting data the images were cropped to the interested area and to reduce size

Data Augmentation:

As the dataset was very small , Data Augmentation was used to increase the size of the dataset. Data augmentation is a technique of artificially increasing the training set by creating modified copies of a dataset using existing data. This also helps overcome over fitting. In the case of this project **Augmentor** is used which is a python package designed to aid in data augmentation. Some Augmentor functions used in this project are:

Figure 3.3.4. shows Original Image:

Functions:

- Zoom()

Figure 3.3.5. shows Zoomed Image:

- Random_Brightness()

Figure 3.3.6. shows random_brightness Image:

- Flip_top_bottom()

Figure 3.3.6. shows flipped Image:

- GreyScale()

Figure 3.3.7. shows greyscale Image

- Rotate90()

Figure 3.3.8. shows rotated Image

- Skew_top_bottom()

Figure 3.3.9. shows top_bottom skewed Image

- Skew_left_right()

Figure 3.3.10. shows left_right skewed Image

Uploading data to google drive:

After data augmentation the size of dataset was increased to 4827 from the original size of 593 images and then this data was uploaded to google drive so it could be used with Google Colab for the training of the model.

Getting rid of unwanted data:

Before uploading the data the images with less than 10kb size were removed manually and After uploading dataset to google drive unwanted images were removed for example the Images which didn't contain the extensions [jpeg, jpg, png].

Train, Test, Validation split:

The data division of 70:20:10 split was used where 70% was used for Training, 20% for validation and 10% for testing.

Building the model:

To build the model first the Sequential model API is imported it is a way of creating deep learning models where an instance of the Sequential class is created and model layers are created and added to it.The second step involves importing the required layers to build the model. In this model 3 convolution layers are used with varying filters e.g (16,32,64) in the first layer the input shape is 256x256 with 3 channels and Relu activation function is used. Activation functions are used to introduce non-linearity in the outputs of neuron layers. After each convolution layer a pooling layer is used to decrease the feature map. After the pooling layer a flatten layer is used which coverts the multi-dimensional input into one dimensional output in order to feed it to the fully connected layer.At the end a fully connected layer is used with sigmoid activation function which is useful for binary classification which in this case is being used.

The next step is compiling the model for which a optimizer is used. An optimizer is an algorithm or function that adapts the neural network's attributes, like learning rate and weights. Hence, it assists in improving the accuracy and reduces the total loss. In this case Adam optimizer is used.in this case

Fit() method is used for training the model on 20 epochs. Epochs is defined as the total number of iterations of all the training data in one cycle for training the machine learning model

Testing the model:

For testing the model a image is downloaded from the internet and then its resized so the labels are in binary form and then fed into the model by using model.predict() method which predicts the label of new data. The result is then calculated by using a threshold value if the value is greater than threshold we predict 1 otherwise we predict 0. In this case threshold was 0.5.

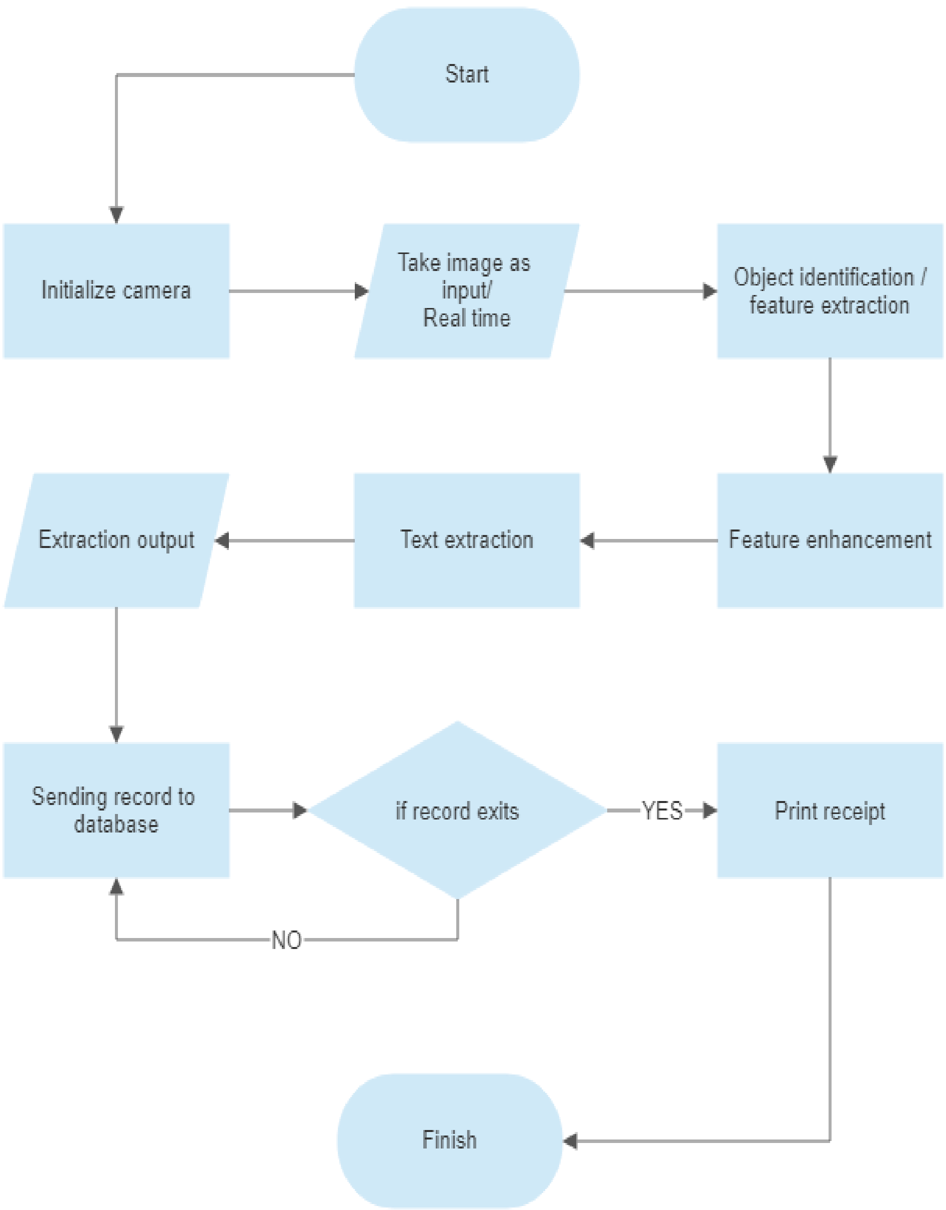
Saving the model for future use:

In the last step we save the model into our desired format to make sure we can use it in the future

3.4. DIAGRAMS

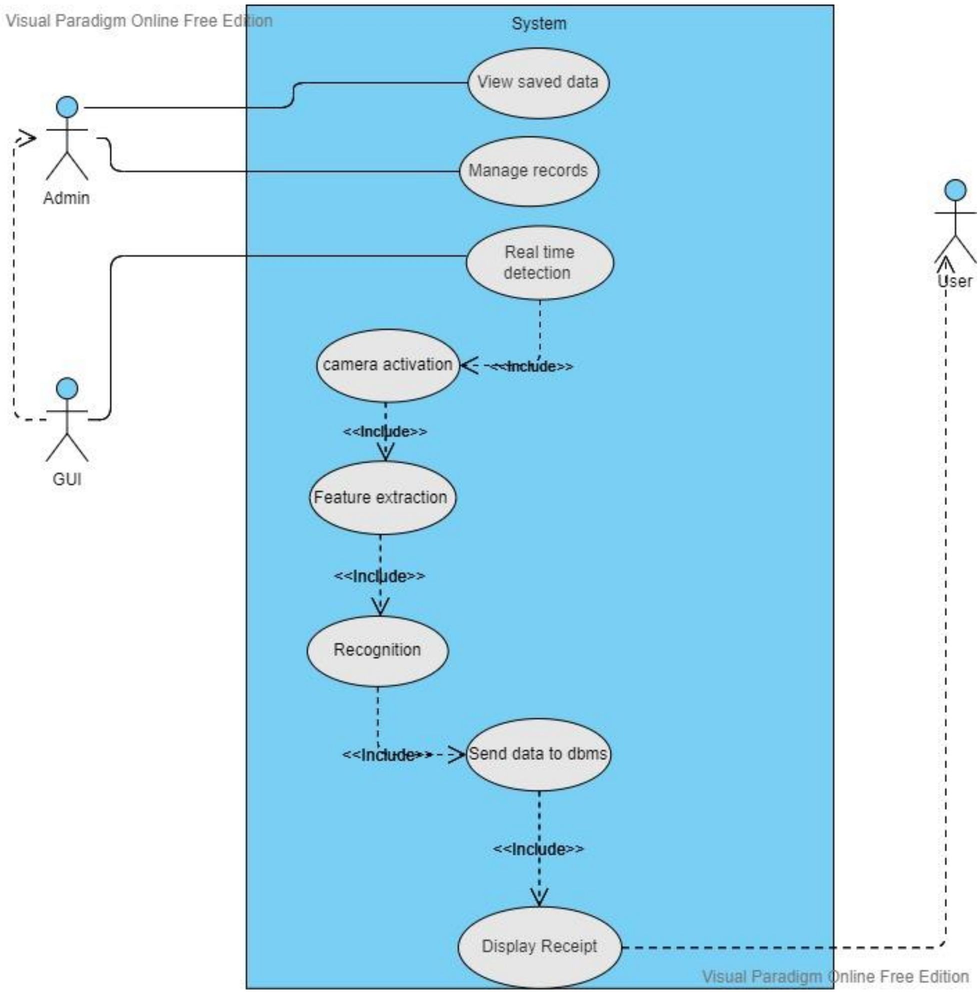
3.4.1. FLOW CHART

Figure 3.4.1.1. shows flow chart



3.4.2. USE CASE

Figure 3.4.2.1. shows use case diagram



3.4.3. CRITICAL PATH

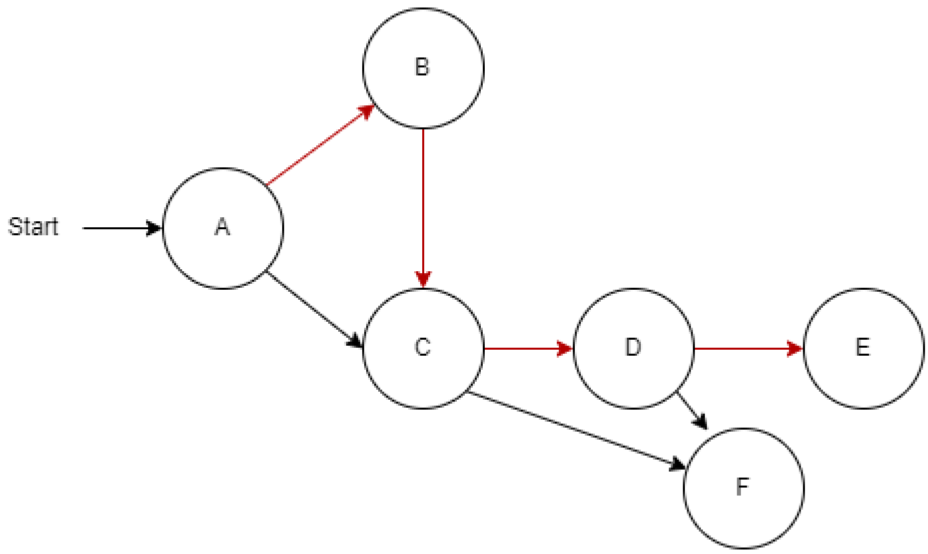
Table 2. Shows Activities

Name	Activity	Predecessor	Duration(month)
Requirement Gathering	A	-	1
Analysis	B	A	1
Design	C	A,B	2
Coding	D	C	3
Testing	E	D	3
Implementation	F	C,D	1

So,

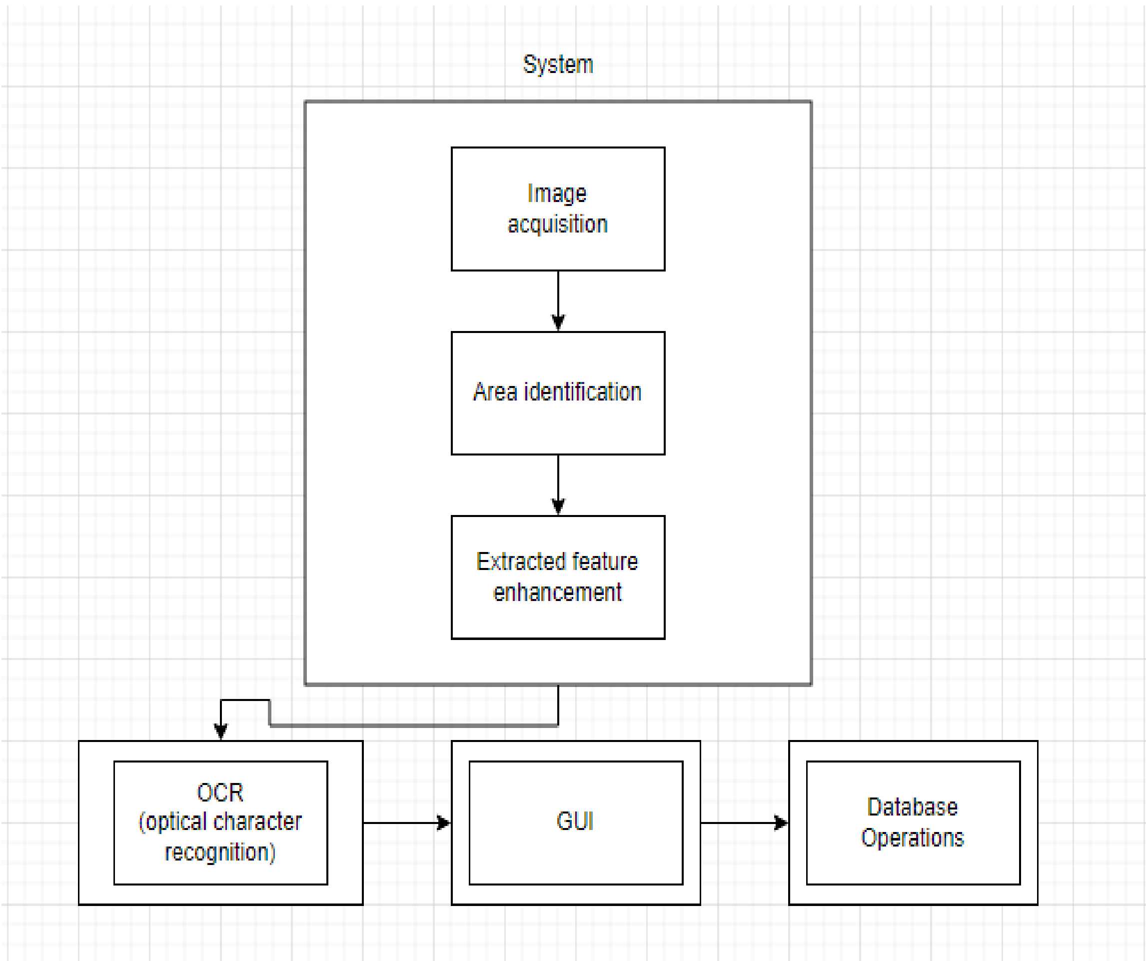
Critical path : A-->B-->C-->D-->E

Figure 3.4.3.1. shows use critical path



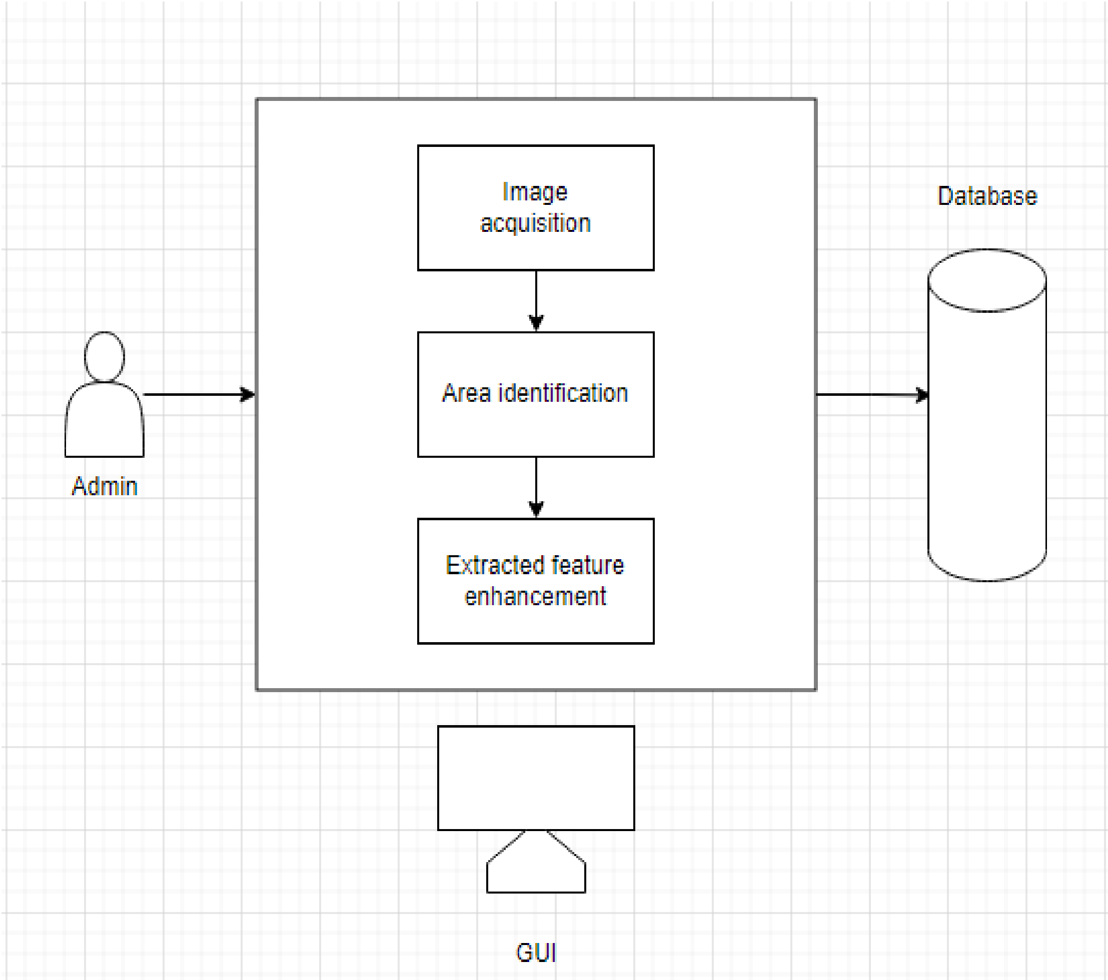
3.4.4. BLOCK DIAGRAM

Figure 3.4.4.1. shows block diagram



3.4.5. ARCHITECTURE DIAGRAM

Figure 3.4.5.1. shows architecture diagram



CHAPTER 4 RESULTS

4.1. Experimental results

Training, Testing and the Validation of the model:

To find out the performance of the model, separate training data, testing data and validation data is created. The training dataset is used to train the model. During training time for checking the model performance, validation set is used which helped tuning the parameters of the model. The test data is used to finding out the performance of final model. The dataset has total 2403 food images. Around 20% of images were used for validation and 70% of images were used to train the model. The remaining 10% were used in for test. During testing, the image was resized to 256×256 pixels.

Model performance:

Subsequent to running 20 epochs the proposed model gained the accuracy of 100% for the training dataset that was created and 99.37% on the validation dataset. Completing the training session the test on random images went pretty well. The model was able perform successfully.The over-all performance of the model is illustrated in **figure 4.1.1 and figure 4.1.2**

Figure 4.1.1. Shows Loss of training and validation

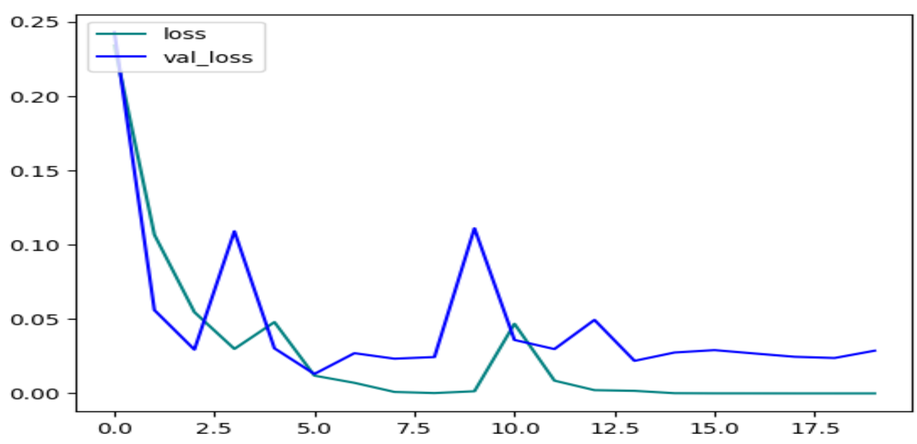
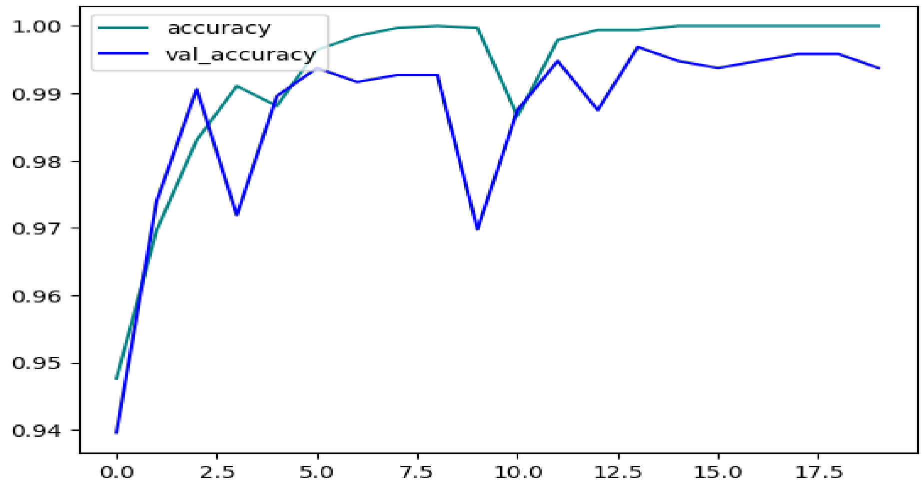


Figure 4.1.2. Shows Accuracy of training and validation



Model Summary:

The visualization of the model summary is given in **figure 4.1.3.** The figure also shows the architecture of the proposed model which includes layers to implement the model. The convolution and max-pooling layers are used in feature extraction part and the dense layer is used as the fully connected layer.

Figure 4.1.3. Shows Model Summary

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_3 (Conv2D)           (None, 254, 254, 16)      448

 max_pooling2d_3 (MaxPooling  (None, 127, 127, 16)     0
 2D)

 conv2d_4 (Conv2D)           (None, 125, 125, 32)      4640

 max_pooling2d_4 (MaxPooling  (None, 62, 62, 32)       0
 2D)

 conv2d_5 (Conv2D)           (None, 60, 60, 64)        18496

 max_pooling2d_5 (MaxPooling  (None, 30, 30, 64)       0
 2D)

 flatten_1 (Flatten)         (None, 57600)             0

 dense_2 (Dense)             (None, 256)               14745856

 dense_3 (Dense)             (None, 1)                 257

=================================================================
Total params: 14,769,697
Trainable params: 14,769,697
Non-trainable params: 0
_________________________________________________________________
```

Outcomes of the System:

In the proposed system, a model is introduced to recognize number plates from images. During this, a machine learning approach has developed to establish the model. In this study, a dataset of images of 2 classes is introduced for recognition. To perform the task ,Convolutional Neural Networks (CNNs) are used which were developed to perform on machine learning approaches. The proposed model was able to get accuracy of 99.37% which proved that the performance of this model is very high. The high accuracy of the model shows that CNN is very suitable for this kind of object recognition.

CHAPTER 5 CONCLUSIONS AND FUTURE WORKS

This paper has presented a number plate detection system for image data captured by smart phone and downloaded from internet using the state-of-the-art detection framework, CNN. The system recognizes 2 types of images Positive (which is the region of interest) and negative. The overall performance of the model to detect number plates is very well and has met a result of about 99.37% accuracy. The applied CNN method is a powerful method for machine learning approaches that successfully recognizes the number plates for the proposed model. The has successfully performed image classification and object detection. In future, the existing dataset will be enhanced with more images. Also future work will involve the integration of the proposed system into a mobile application.

REFERENCES

[1]. Shahare, N., Parsewar, S., Bhange, P., Kshirsagar, D., & Sawwalakhe, R. (2020). Toll tax collecting system using optical character recognition.

[2]. Nagarajan, V., Mohanlal, B., Murali, G., & Gnanasekaran, G. Number Plate Recognition Using CNN For Identification Of Theft In Toll Collection System

[3]. Mandal, A., Singh, D., Shukla, R., & Patil, M. V. IOT and Machine Learning based Automatic Toll Collection System.

[4]. Sun, Y., Xue, B., Zhang, M., Yen, G. G., & Lv, J. (2020). Automatically designing CNN architectures using the genetic algorithm for image classification. *IEEE transactions on cybernetics*, *50*(9), 3840-3854.

[5]. Benjdira, B., Khursheed, T., Koubaa, A., Ammar, A., & Ouni, K. (2019, February). Car detection using unmanned aerial vehicles: Comparison between faster r-cnn and yolov3. In *2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS)* (pp. 1-6). IEEE.

[6]. Miao, Y., Liu, F., Hou, T., Liu, L., & Liu, Y. (2020, November). A nighttime vehicle detection method based on YOLO v3. In *2020 Chinese Automation Congress (CAC)* (pp. 6617-6621). IEEE.

[7]. Lei, X., Pan, H., & Huang, X. (2019). A dilated CNN model for image classification. *IEEE Access*, *7*, 124087-124095.

[8]. Chauhan, R., Ghanshala, K. K., & Joshi, R. C. (2018, December). Convolutional neural network (CNN) for image detection and recognition. In *2018 first international conference on secure cyber computing and communication (ICSCCC)* (pp. 278-282). IEEE.

[9]. Risdin, F., Mondal, P. K., & Hassan, K. M. (2020). Convolutional neural networks (CNN) for detecting fruit information using machine learning techniques. *IOSR Journal of Computer Engineering (IOSR-JCE)*, *22*(2), 01-13.

[10]. Ge, Z., Cao, G., Li, X., & Fu, P. (2020). Hyperspectral image classification method based on 2D–3D CNN and multibranch feature fusion. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*, 5776-5788.

[11]. Chaganti, S. Y., Nanda, I., Pandi, K. R., Prudhvith, T. G., & Kumar, N. (2020, March). Image Classification using SVM and CNN. In *2020 International conference on computer science, engineering and applications (ICCSEA)* (pp. 1-5). IEEE.

[12]. Azimjonov, J., & Özmen, A. (2021). A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways. *Advanced Engineering Informatics*, *50*, 101393.

[13]. Akram, M. W., Li, G., Jin, Y., Chen, X., Zhu, C., Zhao, X., ... & Ahmad, A. (2019). CNN based automatic detection of photovoltaic cell defects in electroluminescence images. *Energy*, *189*, 116319.

[14]. Varshni, D., Thakral, K., Agarwal, L., Nijhawan, R., & Mittal, A. (2019, February). Pneumonia detection using CNN based feature extraction. In *2019 IEEE international conference on electrical, computer and communication technologies (ICECCT)* (pp. 1-7). IEEE.

[15]. Mahto, P., Garg, P., Seth, P., & Panda, J. (2020). Refining Yolov4 for vehicle detection. *International Journal of Advanced Research in Engineering and Technology (IJARET)*, *11*(5).

[16]. Tourani, A., Soroori, S., Shahbahrami, A., Khazaee, S., & Akoushideh, A. (2019, March). A robust vehicle detection approach based on faster R-CNN algorithm. In *2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA)* (pp. 119-123). IEEE.

[17]. Balamurugan, G., Punniakodi, S., Rajeswari, K., & Arulalan, V. (2015, February). Automatic number plate recognition system using super-resolution technique. In *2015 International Conference on Computing and Communications Technologies (ICCCT)* (pp. 273-277). IEEE.

[18]. Patil, V. T., Burud, V. S., Ghewari, P. B., & Desai, P. B. (2020). An Efficient Approach for Automatic Number Plate Identification Using Optical Car Recognition (OCR) Technique.

[19]. Poorani, G., Krishnna, B. K., Raahul, T., & Kumar, P. P. (2022). Number Plate Detection Using YOLOV4 and Tesseract OCR. *Journal of Pharmaceutical Negative Results*, 130-136.