Project and Professionalism (6CS007)

License Plate Detection

Student Id : [2040368]
Student Name : [Akash Chanara]
Supervisor : [Sagar Lamichhane]
Cohort/Batch : 4

Declaration Sheet

Award Title: ***BSc(Hons) Computer Science***

Declaration Sheet

(Presented in partial fulfillment of the assessment requirements for the above award.)

This work or any part thereof has not previously been presented in any form to the University or to any other institutional body whether for assessment or for other purposes. Save for any express acknowledgements, references and/or bibliographies cited in the work. I confirm that the intellectual content of the work is the result of my own efforts and of no other person.

It is acknowledged that the author of any project work shall own the copyright. However, by submitting such copyright work for assessment, the author grants to the University a perpetual royalty-free license to do all or any of those things referred to in section 16(I) of the Copyright Designs and Patents Act 1988. (viz. to copy work; to issue copies to the public; to perform or show or play the work in public; to broadcast the work or to make an adaptation of the work).

Student Name: Akash Chanara

Student ID Number: 2040368

Signature: ..... Date: 2021/04/11

(Must include the unedited statement above. Sign and date)

Please use an electronic signature (scan and insert)

Abstract

As the number of vehicles grows, so does the number of people who break traffic laws. License Plate Recognition is a project that was created to recognize the license plates of light vehicles registered in Nepal's Bagmati Zone. The license plate is in a vehicle image using various image pre-processing techniques such as HSV filter, blurring, thresholding, and edge detection. Then, A connected component mask is used to extract all segmented characters from the plate, which runs in a loop and recognizes the characters from the plate as well as the contours sketched in the mask. The model was created using Convolutional Neural Networks after the characters were extracted. The model, which was created using a Convolutional Neural Network, classified twelve characters with a training accuracy of 98.21%.

Acknowledgement

I'd like to express gratitude to everyone who made it possible for me to finish this project supervisor Mr Sagar Lamichhane deserves special thanks for his unwavering assistance and assistance in coordinating and guiding me.

I'd like to express my gratitude to Mr Nirmal Thapa, the module leader of Project and Professionalism, for his tireless efforts and unwavering assistance in guiding us toward our goal by providing a good report format.

1 Contents

1. INTRODUCTION

1.1 Academic questions:

- What is the best way to use CNN to recognize the characters?
- How accurate is CNN's recognition of the characters on the plate?

1.2 Amis and Objectives:

1.2.1 Aims

The project aims to identify the number plates of vehicles registered in Nepal's Bagmati zone. The primary goal is to detect bike number plate that trains into CNN model and the dataset contains 12 classes of data for character recognition in number plates. A secondary goal is to develop a web application with the following characteristics:

- Upload images to detect license plate numbers.
- Allow users to save recognized numbers to a database.

1.2.2 Objectives

The following requirements were achieved in the development of the proposed model are described below:

- Convolutional Neural Network research, which aided in the system's character segmentation from the plate region.
- To better understand how the system works, make a flowchart and use a case diagram. Recognize the fundamental concepts of license plate character recognition.
- Understand the algorithm that was used to develop a system for recognizing license plate numbers. The number was recognized in the database by Store. Link the artificial intelligence model to the license plate recognition app. Evaluate artificial intelligence's performance.
- Make a report based on a project.

1.3 Brief description of the artefact produced.

The number of vehicles on the road in developing countries is growing by the day. Simultaneously, there is a growing demand for a system that can identify vehicle

plates and numbers. Since the device must create a complex image in an environment where pollution affects the license plate zone, as well as real road conditions and camera deployment angle variance, using cameras to implement a license plate detection system is a difficult process. All these issues make it difficult for a license plate detection system to identify a character, but research teams have proposed using neural networks in license plate detection systems to better position and identify them. (Bharat, 2013)

More than textual data was processed by advanced computer technology to solve everyday problems. Medical, monitoring, control, and engineering domain applications all use optical system inputs. The ability of computers to process images and convert them into something meaningful has grown in popularity. Image processing is also used to monitor vehicle parking systems, vehicle entry into restricted areas, traffic control systems, and highway electronic toll collection, among other things. To do this, the computer must record and process the vehicle license plate number on the screen. To capture the picture of the vehicle and locate the plate area, various techniques are used. The picture is then fed into the computer to be processed further. The performance of the loop is the textual number of the vehicle license plate. The performance is used for car recognition, parking charge, and parking space access authorization in a parking system. (Patel, 2013)

1.3.1 Problem Statement

Due to the rapid increase in the number of vehicles in our country, all parking spaces are currently full. Many business owners use handwritten databases in ledgers to manage their parking spaces. The data in the ledgers is insecure, and the parking lot owner may be hesitant to trust the workers at the parking space because they could commit fraud while entering data into the database in ledgers.

A lot of accidents happen on the road these days, and some of them are hit-and-runs. In the Kathmandu valley, at least one hit-and-run accident occurs every 14 minutes, according to the Metropolitan Traffic Police Division. The driver who caused the accident has yet to be identified. In fiscal 2015-16, there were 338 hit-and-run cases, 612 in fiscal 2016-17, and 522 in fiscal 2017-18, according to the MTPD. People driving on the road are becoming more careless as a result of these factors. Many drivers break traffic laws, and traffic cops can't keep track of everyone on the road to punish three of them. Many vehicles are also stolen from various locations, but they are extremely difficult to locate because it is impossible to inspect each automobile. (Times, 2020)

1.3.2 The project as a solution

The proposed system can retrieve a number from a vehicle's number plate, and it will help people solve all of the problems listed above. This system saves the number and time when it recognizes a character from the plate in a database. Traffic cops

can use this system to track down stolen vehicles and to check the number plates of traffic rule violators in order to fine them.

1.4 Scope and Limitations

The proposed artefact is intended to read characters from light vehicles that have been registered in the Bagmati zone. As we all know, Nepalese vehicles have license plates on both the front and back sides. The front license plate, on the other hand, is the subject of this artefact.

This artefact has the following limitations:

- Due to a lack of data, this artefact has only been trained to detect characters from vehicles registered in the Bagmati zone and classified as light vehicles.
- Only works for the license plate on the vehicle's front bumper.
- Is unable to read the characters on the new number plate that was issued after the country was divided into Pradeshs (States).
- The characters from the embedded number plates are not recognized.

1.5 Structure of the report:

The following is a brief overview of the topics covered in this report:

- Literature review: A brief overview of artificial intelligence, as well as some of the algorithms and image preprocessing techniques used in this project. Several other projects of a similar nature are also investigated and analyzed.
- Main Points: The fact-finding techniques and System Requirement Specification for the proposed system are presented first. The development methodology, UML diagrams, database design, UI/UX design, and architecture design are all included in this section. Testing is also done to ensure that the requirements of the Web application are met, using a variety of testing approaches and methods.
- Academic questions: Here, the questions labelled as such are answered and briefly discussed.

1. Conclusion: This section discusses the final report and the artefacts produced, as well as improvements that could be made to the project.

2 Literature review

Image classification becoming increasingly popular, and it gets even developed fashionable. It is capable of recognizing and identifying images in the same way that humans are. Because it recognizes all of the tagged faces on the site, Facebook is a great example of image classification. Image classification is used in a variety of industries, including e-commerce, healthcare, and gaming. Deep learning is one of the methods for image classification. It falls under the category of Artificial Intelligence (AI). A machine that can act and think like a human is referred to as AI. Machine learning, which is similar to deep learning, is another commonly used system for image classification. Machine vision is another technique used in image classification. It aids in the recognition of people, places, objects, actions, and writing in photographs. (Abu, 2019)

2.1 License Plate Recognition Sub-process

The overall system for license plate recognition is divided into four sub-processes. These sub-processes are image preprocessing, region plate localization, plate number extraction, and character detection, and are described in detail below:Image Pre-processing

For any image processing system to achieve the required result, the pre-processing of the image is a great step. The input image needed to be pre-processed in order In this mission, to carry out the number plate localization. Thus for HSV filtering, blurring and threshold tasks, the input image was added. The managed picture then develops prepared for the number plate to be localized.

- HSV

HSV model is specified as colours. Often, the HSV colour model is preferred over the RGB model in situations where the importance of colour plays a vital role in the processing of images. Depending on the car ownership, Nepalese license plates have various coloured plates. For example, private automobiles have red-coloured plates with white number, and government automobiles have black plates with white number. Masking is then achieved depending on the colour of the number plates of the vehicles. (Ashok Kumar Pant, 2015)

- Thresholding

It simply implies converting a picture into black and white. In this case, If the value exceeds the value of the pixel, the threshold value, one value is allocated or another value (maybe black) is assigned. An adaptive thresholding, the algorithm regulates the threshold value for a specific image region. So, for different regions, We get same

image at different levels, which allows us to produce better results for images with varying lighting. The value of the Gaussian threshold in the Gaussian threshold where the weights are a Gaussian window is the weighted sum of the neighborhood values. (Mordvintsev, 2013)

- Sobel edge detection

The dimensional spatial gradient on an object and provides the edge-related high spatial gradient regions with additional significance. It is used to find the magnitude of the gradient at each point in a greyscale picture. This approach helps to strengthen the element's edge on both sides, which makes it possible for the edge is thick and light looking. It also provides a soothing effect on the random noise of the image. (Radhika Chandwadkar, 2013) I used vertical edge and horizontal edge detection kernels for Sobel edge detection.

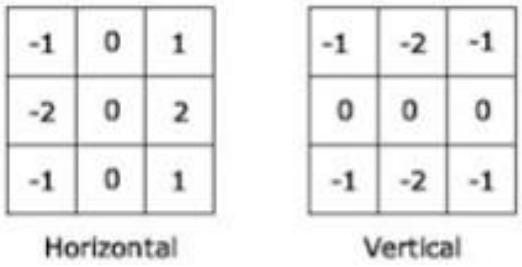


Figure 1- Sobel Edge detection (O. R. Vincent, 2009)

2.1.1 Localization of the plate region

To perform the picture preprocessing task, we need to locate the plate from a vehicle image. After the image is preprocessed and ready, the contour recognition system is used to perform the task of localization. It is probable to define contours as a curve connecting all unceasing points that have a similar colour or strong point. Contours are a helpful instrument for the study of form and the recognition and identification of objects. (OpenCV, 2020)

2.1.2 Character extraction

The plate picture is converted into grayscale once the location of the plate is completed, and erosion has been applied to the picture. The procedure for erosion causes boundary pixels to be lost to object. The attrition process increases the numeral of pixels with the zero (background) value and reduces the number of pixels with the first (front) value. The net effect in a picture sharpens the image as the unwanted pixels are 'erased'. The erosion technique is similar to high-pass filter sharpening used in linear image filtering.. (Ravi S, 2013)

2.1.3 Character recognition

The Convolutional Neural Network is used in this project to train the model to recognize characters.

1. Convolutional Neural Network

Convolutional neural networks are one of the most common Deep Learning architectures for image recognition. It gained popularity because, when compared to Artificial Neural Networks (ANN), it has fewer trainable parameters and can detect complex features at deeper layers that ANN can't. (Saad ALBAWI, 2017). CNN has a 99.77 percent accuracy rate for predicting MNIST handwritten digits and a 97.6 percent accuracy rate for classifying more than 10 objects on 5600 images. Not only has CNN outperformed other machine learning or deep learning algorithms in classifying images, but it has also outperformed humans in classifying bird and dog species. (Samer Hijazi, 2015). The term Convolutional Neural Networks comes from a mathematical procedure called convolutions. In a CNN, a filter matrix in the convolutional layer convolves the image, which is then followed by an activation function, pooling layers, and fully-connected layers. (Saad ALBAWI, 2017)

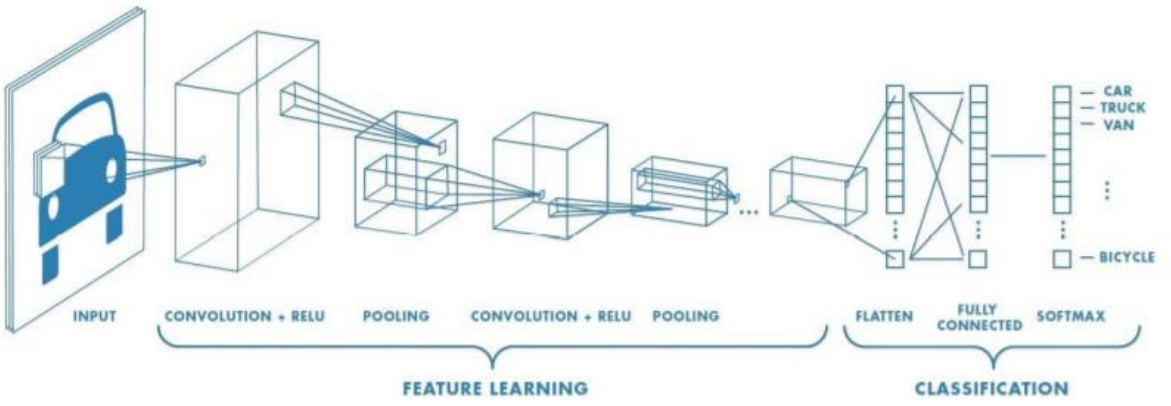


Figure 2-Convolutional Neural Networks (CNN) Architecture (Saha, 2018)

Convolution is performed using trainable filter matrices of different sizes in the first layer of a CNN, which takes an image as input (3x3, 5x5, etc.). The filter multiplies the pixels of a filter image and applies the multiplied numbers to produce new pixels or image features in a convolution layer. The filter depths are determined by the image channels; for example, the filter size for an RGB image is 3x3x3, 5x5x5, while the filter size for a grayscale image is 3x3x1, 5x5x1.To convolute every pixel of an image, use strides to move the filter slides horizontally and vertically. Strides are numbers like 1, 2, and so on that determine how many rows or columns are used to convolute images pixels vertically and horizontally when filtering slides. (Saad ALBAWI, 2017)

After image convolution, these feature pixels, also known as feature maps (LeCun & Bengio, 2017), are transferred to non-linear activation functions like ReLU, sigmoid, tanh, and others. The activation functions' outputs are then transferred to the pooling layer, which decreases the size and volume of feature maps. A 2x2 matrix slides around the feature maps with strides of 2 in the pooling layer to achieve max-pooling, which removes the most feature maps from a 2x2 matrix. After the first convolution and pooling layer, an image's height and width are decreased, while its depth increases and is proportional to the number of filters used in that layer. When a 32x32x3 RGB image is passed through a convolution layer with 16 3x3x3 filters and one stride, the result is a 30x30x16 image. In the pooling layer, our final features maps will be 15x15x16 with a stride 2 matrix of 2x2x16. (Saad ALBAWI, 2017).

CNN has several convolutional layers accompanied by a pooling layer; the more complex features CNN detects, the more convolutional layers it has. After the output of the final convolutional layer, feature maps are flattened to obtain a feature vector, which is then transferred to fully-connected layers or artificial neural networks. To

classify an image, a completely connected layer is used. Since CNN only transfers feature values to fully-connected layers, it is considered the best algorithm for image recognition because it can extract complex features and minimize noise from an image. The filter matrix elements are also learnable parameters, and the filter is modified as training progresses until it can correctly detect features from an image. (Saad ALBAWI, 2017)

1. Maximum Pooling Layer

The spatial resolution feature maps are minimized. It also aids in the attainment of spatial invariance through the use of pooling layers, as well as input modification and conversion. Between the two pooling layers, In the models average pooling and maximum pooling layers, the maximum pooling layer has been removed. Full pooling is used to spread the maximum value inside the receptive area to the next layers. (Zenghui, 2017)

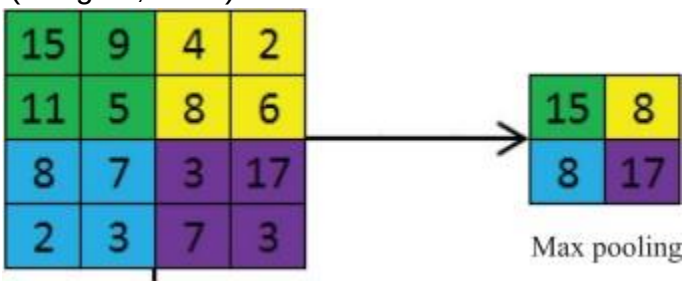


Figure 3 -Maximum Pooling layer work (Zenghui, 2017)

2. Batch normalization

Normalization of the input layers can be accomplished by changing and scaling the stimulations. The hidden layers, on the other hand, are normalized using batch normalization. The amount of value that shifts around the hidden unit values is reduced when batches are normalized. Using batch normalization, the layer can read a little more data on its own. subtracting the batch average and dividing it by the batch's average deviation It is possible to normalize the output of the previous activation layer to increase the neural network's stability. (towardsdatascience, 2020)

3. Fully Connected layer

The output is Flattened or converted final convolutional layer's characteristic maps into a one-dimensional layer number array then linked to the dense layers that are more fully linked. The characteristics gathered Reduced from the convolutional layers are in this layer. Each dense layer is linked to a ReLU activation function that takes In the final fully-linked layer, only the positive value and the number of output nodes are equivalent to the number of class. (Togashi, 2018)

1. Binary Crossentropy

The Binary Crossentropy loss function is used in binary classification tasks where the real values are either 1 or 0. It determines the logistic loss, also known as the negative logarithmic of the predicted value. For exampleIf the Ni sample has one actual value, log(p(yi)) is added to the loss function; if the actual value is zero, log(1 - p(yi)) is added to the loss function. The expected value has a probability of 1 (p(yi)). (Daniel, 2018)

$$H_p(q) = -\frac{1}{N}\sum_{i=1} y_i \cdot log(p(y_i)) + (1 - y_i) \cdot log(1 - p(y_i))$$

Figure 4-Binary Cross-Entropy function (Daniel, 2018)

2. Categorical Crossentropy

Definitions of categorical words When a classification problem has several groups, crossentropy is used as a loss function. It also employs the binary classification cross-entropy function. (Daniel, 2018)

$$-\sum_{c=1}^{M} y_{o,c} \log(p_{o,c})$$

Figure 5-Categorical Cross-Entropy Function (Daniel, 2018)

4. Loss function

The Loss Function is one of the most critical aspects of Deep Learning. It measures the difference between the expected value of the neural network and the input's real value, and the input's actual value The Loss Function measures how close the predicted value is to the actual value, as well as how accurate the predicted value is. (Himanshu Jain, 2016). The loss function value decreases as the neural network's accuracy increases. The main aim of a neural network is to minimize the loss function. (Brownlee, 2019). The following are some of the loss functions:

5. Activation function

To detect complex patterns or features, In hidden layers and neurons in the output layer, Deep Learning architecture uses a non-linear function. As a result, the activation function transforms linear functions created from input features, weights, and bias into non-linear functions, allowing the algorithm to learn complex and interesting functions. The activation function is one of the most important research topics in Deep Learning because it specifies the time needed to train a deep neural network with higher accuracy. (Forest Agostinelli, 2015)

Some of the most widely used activation functions are described below:

1. Rectified Linear Unit (ReLU) function

Nair and Hinton's paper in 2010 popularized the ReLu activation function. The ReLU function has been discovered to not only speed up neural network training but also to make the neural network more generalized. If the input is less than 0, the ReLU function simply converts it to If the input is greater than or equal to 0, it returns the same input number. Because of its simplicity, the ReLU function calculates gradient faster than the sigmoid and tanh activation functions during backpropagation. ReLU solves the problem of vanishing gradients that can occur while using sigmoid and tanh in hidden layers since it is similar to a linear function. ReLU has been used in hidden layers with sigmoid or softmax activation functions in the output layer for classification problems.

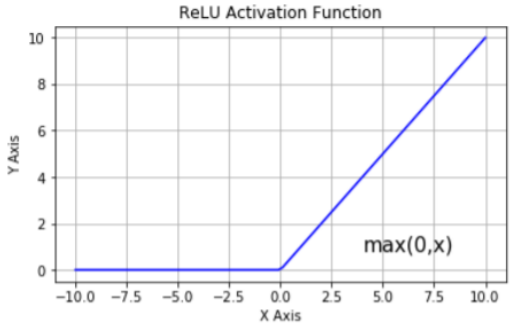


Figure 6- ReLU Activation Function graph (Szandała, 2017)

2. Softmax function

For multi-class classification tasks, The Softmax Activation Function is commonly used in a neural network's output layer. The Softmax feature in the output layer transforms the logit layer's results (before proceeding to the activation function) into probabilities that are distributed to each output layer neuron. In the logit layer, it distributes the probabilities of those three neurons for a three-class classification task. As a result, adding the outputs of those three neurons' softmax functions equals one. It's mostly used in Deep Learning's output layers when there are more than two classes in a classification task (Uniqtech, 2018)

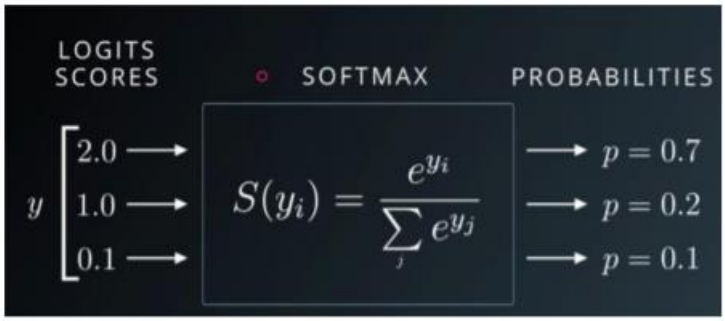


Figure 7-Softmax Activation Function

3. Sigmoid Activation Function

Prior to the use of ReLU in neural networks, The first preference activation function was the Sigmoid Activation Function, also known as the Logistic Function. It compresses the data into a 0.0 to 1.0 size. The number of inputs greater than 1.0 is reduced to 1.0.and reduces the number of inputs less than 0.0 to 0.0. (Brownlee, 2019). For binary classification tasks, In the output layer of neural networks, sigmoid is now used. (SHARMA, 2017)

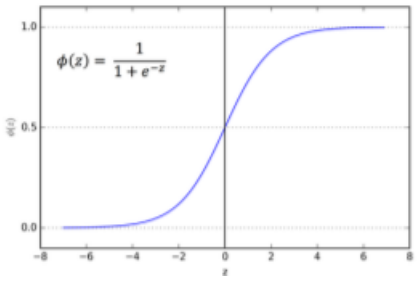


Figure 8-Sigmoid Activation Function graph (SHARMA, 2017)

Dropout layer

A technique for preventing overfitting and providing a way exponentially and efficiently Combine the various architectures of the neural network called a dropout. For a temporary period, the dropping layer eradicates a part and connections to the

neural network. The part that is released is randomly designated. (Geoffrey Hinton, 2014)

2.2 Related works

During the research found that Automatic License Plate Detection used Indian license plate and image processing and Neural Network is done In this system, a picture was used to locate vehicle number plates from the structural processing of a vehicle after pre-processing, However, it offered a poor recital within a bright setting, so an edge detection algorithm that showed significant results was studied and replaced. Using morphological filtering, the accuracy obtained was 83%, while Using upright and horizontal edge detection approaches, the precision was 100%. To predict the number, a feed-forward backpropagation algorithm was used. The neural network has been optimized to gain better predictive accuracy.

It was found that the accuracy also increased when the number of hidden layers was raised. (P. Surekha, 2018)

Method	Total Samples	Plate Localization	Efficiency (%)
Morphological Processing	105	87	83
Horizontal and vertical edge projection	105	105	100

Figure 9- Accuracy of localization using distinct methods. (P. Surekha, 2018)

P.Meghana, S.Sagar Imambi, P.Sivateja, and K.Sairam released Image Recognition for Automatic Number Plate Surveillance in IJITEE in 2019. They started by converting the RGB picture to greyscale and reducing noise with median filters. Vehicle license plates were located using edge detection. The accuracy of the number plate localization using the edge detection technique was 93%. Using a histogram approach, The image's pixel count was plotted as a brightness wave. OCR was used to recognize the characters, and Template matching was used to recognize the characters. (Meghana, 2019)

Elsevier Ltd released a paper by H.Erdinc Kocer and K.Kursat Cevik on artificial neural networks-based vehicle license plate recognition. It was the first year of the new millennium, and the year was 2010. In this project, 259 vehicle images were used. A 220* 50-pixel license plate region was created using image processing algorithms. A combination of The characters were extracted using a combination of clever edge detection and the blob coloring process. The blob colouring algorithm is extended to the binary coding license plate to extract the characters. With the aid of

clever edge detection and the blob coloring algorithm, they were able to achieve a number plate localization accuracy of 98.45 and a number plate character extraction accuracy of 98.82. Two ANN algorithms were used to avoid the difficulty of identifying identical numbers and alphabets such as o and 0, B and 8, and Z and 2. The Artificial Neural Network was trained using a feed-forward back-propagation algorithm, and the characters were classified using a multi-layered perceptron (MLP) ANN. Using two ANN, they were able to obtain a character recognition accuracy of 98.17 percent. (Kocer, 2010)

The research found that a Recognition system for license plates using neural networks and multithresholding techniques was published in 2013 by the International Journal of Computer Applications. The following project uses various multithresholding techniques and the Artificial Neural Network to develop a License Plate Recognition System. First, the image is pre-processed and converted into an image on a greyscale.

The unwanted noise was removed from the image by using median filtering, which helped improve the rate of recognition. The grey-scale image in the segment of the image was subjected to a multi-threshold technique. To remove the segmentation issue, a Blob removal technique has been employed to separate characters and numbers. For character recognition, an artificial neural network was used. The following work has achieved a total accuracy of 98.40%.. (Bharat, 2013)

A study on Automatic platform recognition for Nepali number with vector support equipment was conducted. This paper shows that character recognition for A light vehicle backplate has been done. The first is the pre-processing of the image carried out in the project and the Plate is located using the recognition of contours. After the de-skewing process, Analysis of character projection was done using for segmentation to enhance the accuracy of character segmentation. The support of the Vector Machine was used to train the model to recognize characters from the plate and the Oriented gradient histograms have been generated for prediction from each class. The accuracy of 6.79 % of the system's common errors was assessed using the following model. Typical exactness was 87.59 percent, recall was 98.6 percent, and the f-score was 92.79 percent. The accuracy with which number plates are identified seems to be influenced by precision of dissection and size, resolve, position and lighting of the characters' characteristics. The total precision of this project was 75%. (Ashok Kumar Pant, 2015)

During the research found that the KNN algorithm is used and in 2016 KNN-SVM create a KNN-SVM hybrid model for the recognition license plate of Iranian. In the project, the extraction of features from the characters was converting the standard segmented character into the feature vector, which was 32 * 30 pixels. Therefore, in this study, a hybrid machine was used to identify the characters of the Each license platform requires specific vectors for each machine (KNN and SVM). For the recognition of dissimilar characters, structural features and The crossing count

histogram features of horizontal (H) and vertical (V) were used to abstract features of the types and suckled KNN, while partitioning was to extract the characteristics for alike numbers and suckled into SVM. The precision in the mining of the plate area was 96.01%, the segment of the fonts from the plate 95.24% and the accuracy to be recognized as similar was 96,82%, while the accuracy of the dissimilar character was recognized at 97.31% (Cavus, 2016)

The research found that for the localization of their number plates, HSV filters, colour edge detection and fuzzy techniques of aggregation were used.1088 dissimilar images, 23 pictures futile to locate vehicle's plate numbers. Optical Character Recognition was used for the recognition of numbers from the plates. The success of The OCR prediction was 97.9%. The Overall algorithm accuracy is 93.7%. (Li-Shien Chen, 2004)

Literatures	Localizing License Plate	Segmenting Characters	Recognizing Characters	Accuracy
Elihos	SSD	SSD	SSD	73.3%
Rahman	-	-	CNN	88.67%
Angara	Canny Edge Detector	Vertical Projection	CNN	92.36%
Pant	HSV masking with Sobel Edge Detector	Vertical Projection	SVM	93.2%
Kocer & Cevik,	Canny Edge Detector	Blob Coloring	ANN	98.17%

Table 1-Comparison of found Related work

2.3 Similar System

Recognizer Plate

Plate Recognizer is software for automatic recognition of the number plate, where we can upload the vehicle picture and recognize the character from the show some candidates the plate and even from the platform for the right numbers.

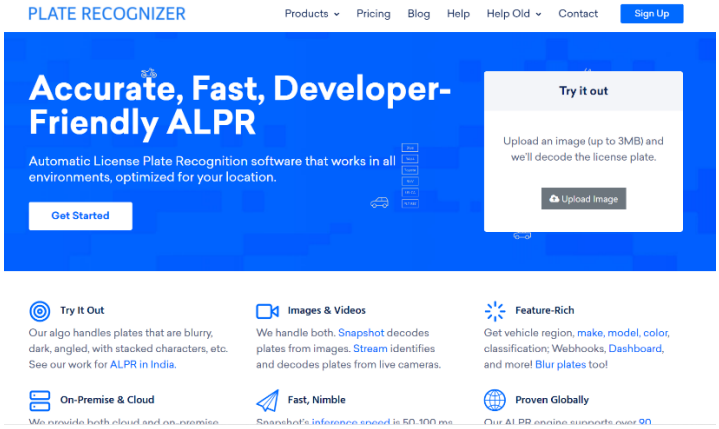


Figure 10- Home page of plate recognizer (platerecognizer, 2020)

After the picture of a vehicle is uploaded, the page displaying the characters on the plate is shown. (platerecognizer, 2020)

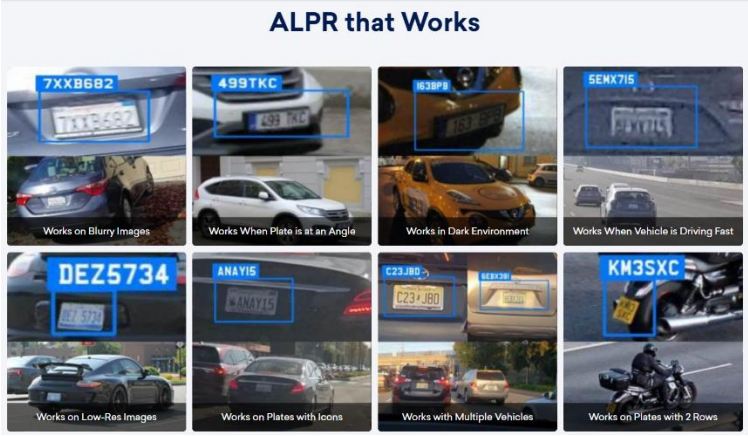


Figure 11- Plate recognizer work

Mobile LRP

Mobile LPR is the identification of a mobile license plate software application. We can choose different license plate set-ups from different countries here, and the plate is recognized by this app. A mobile phone that supports video is needed for this app. We're able to use our cell phone and press the Start button to scan. on the front page of the application and the device opens the camera and the application starts

searching for the vehicle license plate once we click on the start button and the number on the plate is displayed as shown below.

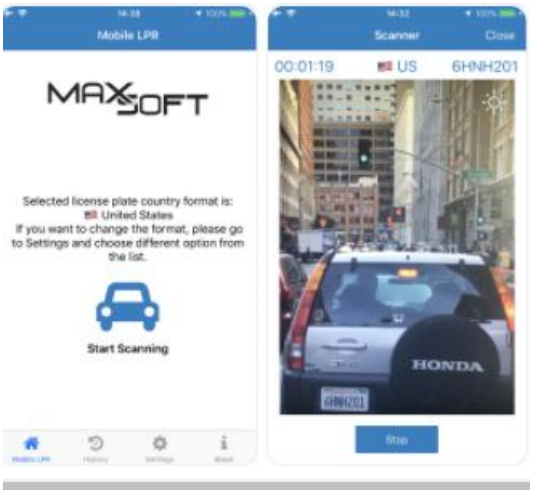


Figure 12-Home page of Mobile LRP

Analysis of similar applications

When comparing features of related applications, the proposed device resembles Plate Recognizer rather than Mobile LPR since the user uploads the image to Plate Recognizer on the internet, while Mobile LPR is an android app that scans the license plate from a car using video and the in-built camera. Both schemes struggle to identify Nepalese license plate characters, but the proposed system aims to develop a model that does.

3 Main Content

3.1 Fact-Finding Techniques

The process of gathering data and information for the analysis of a system's actual requirements using various techniques such as research, surveys, and interviews is referred to as fact-finding. Observation, questionnaires, and interviews were used to gather information. The following are some of the techniques used to gather information for this project:A sampling of existing documentation

Any system can benefit from gathering facts from existing documentation. We can use various types of flowcharts and diagrams to determine the best requirement for our applications by analyzing various types of flowcharts and diagrams from the documents.Research and Site visits

We'll need a way to fulfil the requirements once we've discovered the requirements for developing a system using the above fact-finding technique. Different problems arise from time to time in such a process, and the Internet is a hotspot for finding solutions to such problems. We can find people who have faced similar issues and learn how to solve them, saving us time.

3.2 Functional Decomposition Diagram

The various components of a system, as well as their hierarchical relationships, are depicted in a diagram of functional decomposition A mechanism is depicted in this diagram, which is shown in a top-down style. A functional diagram of decomposition depicts the main function or mission, as well as the sub-functions or tasks needed to accomplish the ultimate objective. The steps involved in breaking down a computer, process, or system operation into its constituent parts are referred to as "functional decomposition" by engineers. A functional decomposition diagram will be used to define the functions – tasks and sub-tasks – as a result of the analysis – and how they interact. The diagram can also be used to talk about and propose solutions to problems. (Kenton, 2019)

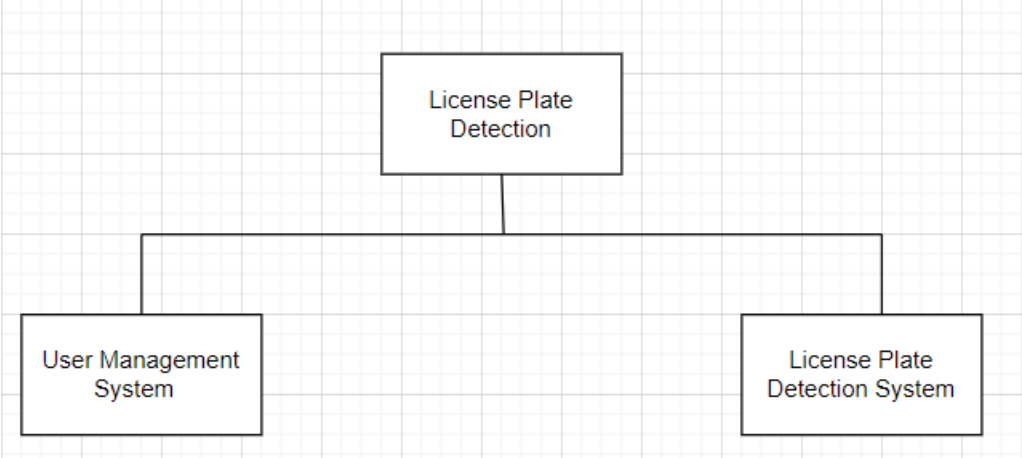
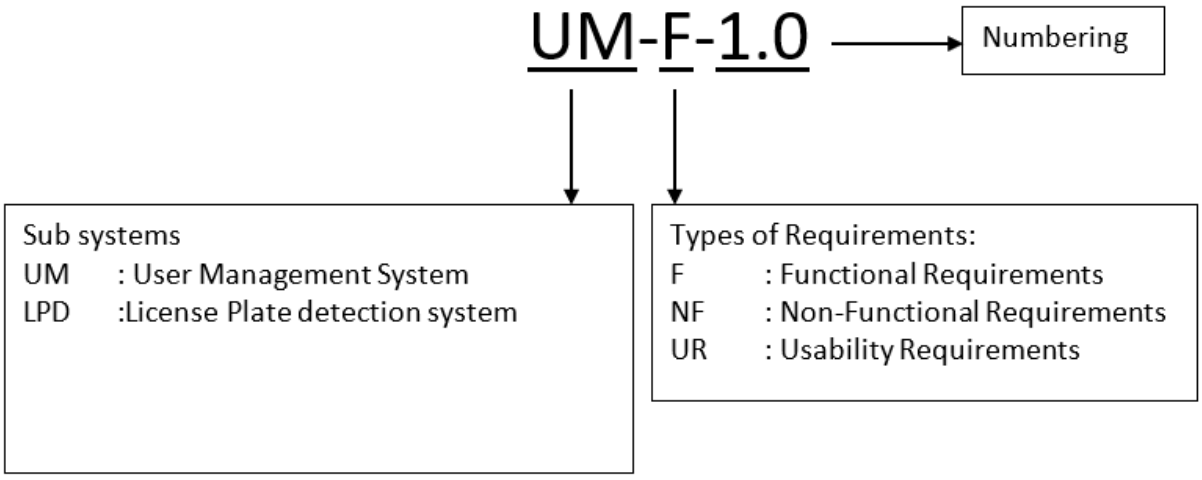


Figure 13 : (FDD)

Legend



User Management System

Req. Code	**Req. Desc**
UM-F-1.0	User should self-register
UM-F-1.1	The registration data need be encrypted before sending it to the server. (security)
UM-NF-1.2	minimum password must be 6 characters.
UM-NF-1.3	‘@’must be in Email
UM-NF -1.4	Password must be a match
UM-F-2.0	Users should allow uploading image

License Plate Detection System

Req. Code	**Req. Desc**
LPD-F-1.0	The system should be able to detect the license plate number on the image
LPD-NF-1.1	The size of the image must be 3MB
LPD-NF-1.2	User should be able to see their license plate number

3.3 Methodology

An incremental model is a non-monolithic model. The incremental model blends the waterfall model's sequential features with the concept model's iteration features. The second increase builds on the first, and any increase can be carried out using either a waterfall or a rapid design model. This model gives users access to software that is divided into multiple periods and allows them to use some of the basic functions while developers work on new product features. (Guru99, 2020)

This project was developed using the incremental or iterative methodology as a software development methodology. This methodology divides the project into iterations, with each module being developed separately. All of the iterations are combined into a single project once all of the modules have been developed. This methodology may be the best fit for developing the proposed system since the prerequisites and skills are well-defined. It's a software development technique in which the planned, implemented, and tested phases are upgraded and/or expanded before the final product is ready.

Why this methodology?

- Methodology aids in the gathering of opinion, which is then used to inform the next stage.
- This methodology is best suited for this project because the requirements may change over time.
- This model allows for more flexibility, requires less information, and allows for faster delivery of the system.

3.3.1 Phases of Incremental model:

1. Requirement phase:

In this step, the basic requirements are evaluated and defined of the incremental model. Gathering the project's requirements and goals resulted in a meeting with the supervisor and a proposal for the proposed system. During this phase, data from the classification model was collected and various related projects were investigated.

Design and Development phase:The device functionality is planned and the implementation process is successfully completed in this phase of the incremental SDLC model. When new functionality or a new version of the product is needed, the incremental model uses the design and development phase. During this phase, the proposed system's Artificial Intelligence model, as well as the UI/UX, were developed.Testing Phase:

Various tests were carried out in this phase during various iterations. This phase included tests for number plate localization, character extraction, and character recognition, as well as accuracy testing.

2. Implementation Phase:

The incremental model's implementation step helps the coding to finish the required improvements to the framework under construction. During the design and development process, the framework was programmed, integrating the user interface with the artificial intelligence model, and feature testing was performed during the testing phase. At the end of this process, the framework is completely developed and ready for clients or users to use 4 Artificial Intelligence (AI) libraries.

Libraries	Details
ImageDataGenerator	To create an image for the purposes of training, testing, and sampling.
NumPy as np	To carry out mathematical calculations.
Sklearn	To create a classification report and a matrix of confusion
cv2	to recognize the individual's face
Matplotlib	images to be printed
Math	To demonstrate some examples of predictions
pickle	To save the model as a.pkl file, follow these steps.
Sequential Conv2D MaxPooling2D	To integrate our CNN model with two convolution layers with ReLU activation function and Batch Normalization, we used a sequential model, which was followed by a MaxPooling layer to reduce the complexity of the stacked models three times.
Dropout	To keep our network from becoming overfit.
Flatten	To feed the output into a fully connected layer after flattening it.
ModelCheckpoint	To determine where the model weights should be checked.
load_model	To load the model for testing purpose

Table 2-Libraries used

3.4 Introduction to Dataset

On GitHub, the dataset used could be downloaded for free and was intended for academic use only .The dataset's images were divided into 12 folders, each corresponding to the number of classes available. The information was gathered from private vehicles registered in the Bagmati Zone, and the classes were 0-9, ba, and pa. There were three sections to the dataset: Training, testing, and sample predictions are all part of the process. Out of a total of 1868 images, 1408 were used for training, 412 for testing, and 48 for sample prediction. The lack of a dataset for image classification and the variety of license plate styles , license plate recognition in Nepal is difficult.

Dataset	Total classes	Training image	Test or Validation image	Sample prediction image	Total images
Characters	12	1408	412	48	1868

Table 3- Information about dataset

3.5 System Design

WorkBreakdownStructure

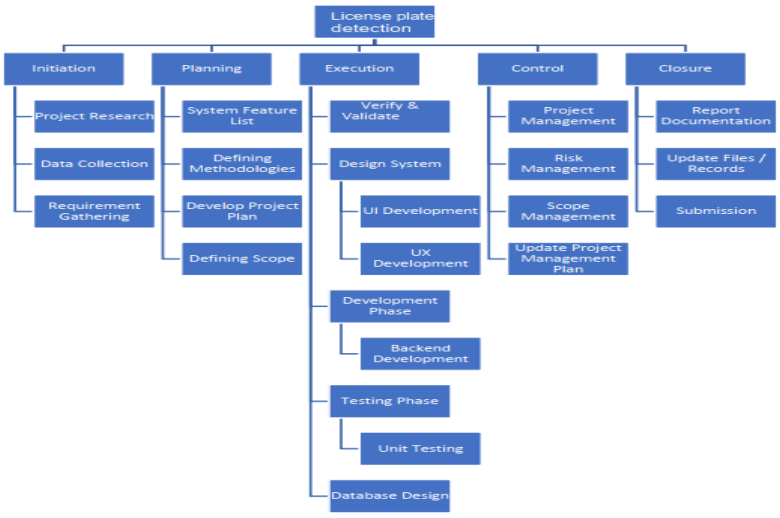


Figure 14-Workbreakdown structure

Business Process Modeling (BPM)

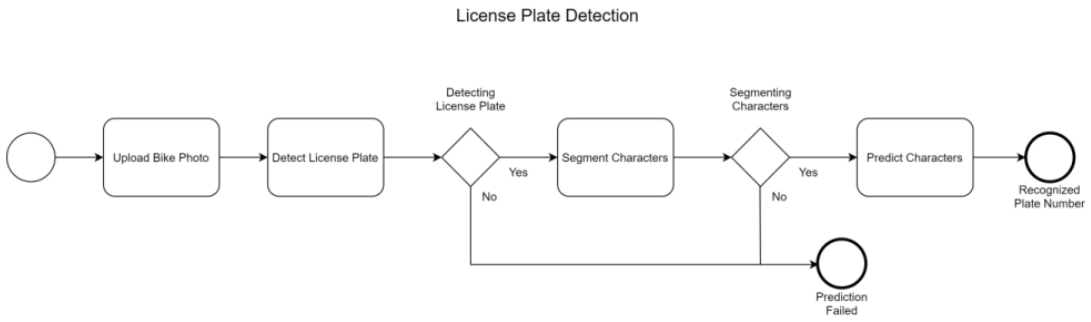


Figure 15Figure 15 (BPM)Business process Modeling

Flow Chart

An online tool called draw.io was used to create the design for the machine learning model.

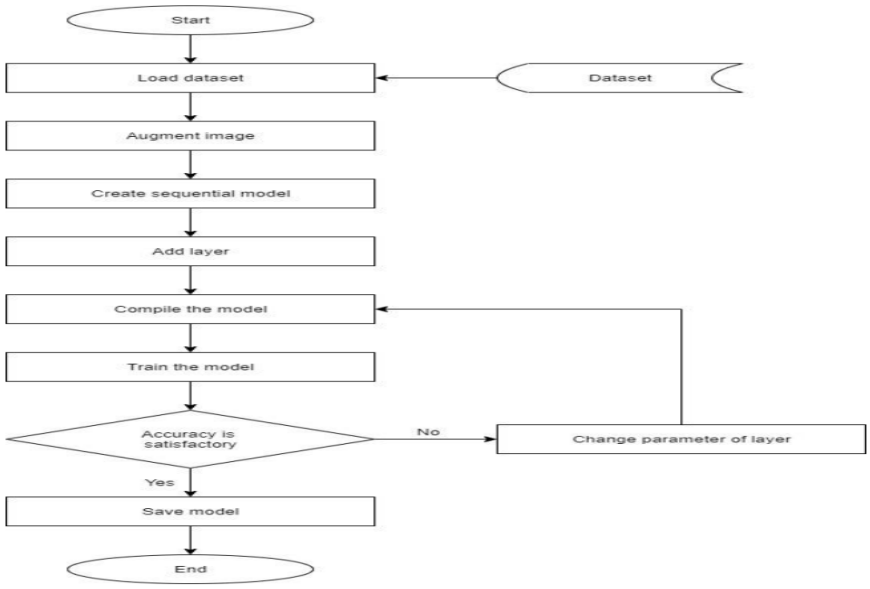


Figure 16- Prediction Engine algorithm

Several actions were taken in the development of a model that predicts the license plate's character. The model was saved using the model after it had reached a satisfactory level of accuracy. To integrate the model with the User Interface, the pickle and save functions were used.

The model was used for prediction after the model accuracy was satisfactory, and the proposed system's working design is shown below.

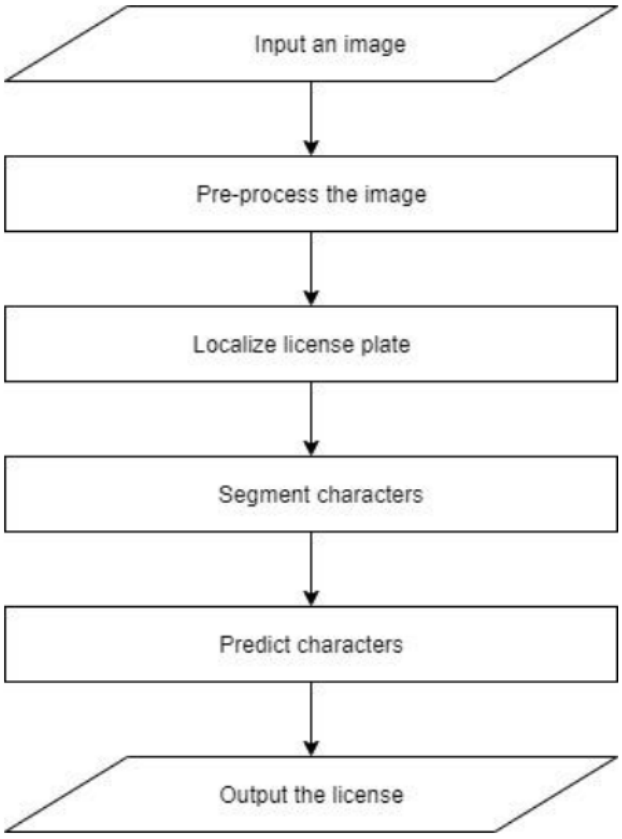


Figure 17-Working system Algorithm

3.6 Unified Modelling Language diagram

Make use of a case diagram to assist you. In the Unified Modelling Language, a use case diagram is a diagram of processes or behaviors. In use case diagrams, actors and use cases are used to model the system's functionalities. A collection of activities, programs, and functions that the device needs are referred to as use cases. Actors are people or organizations that play various roles in the scheme. Case diagrams can be used to visualize system functional requirements that are translated into design decisions and development objectives. It also aids in the identification of any internal or external factors that may have an impact on the system and should be taken into account. (logo, 2020)

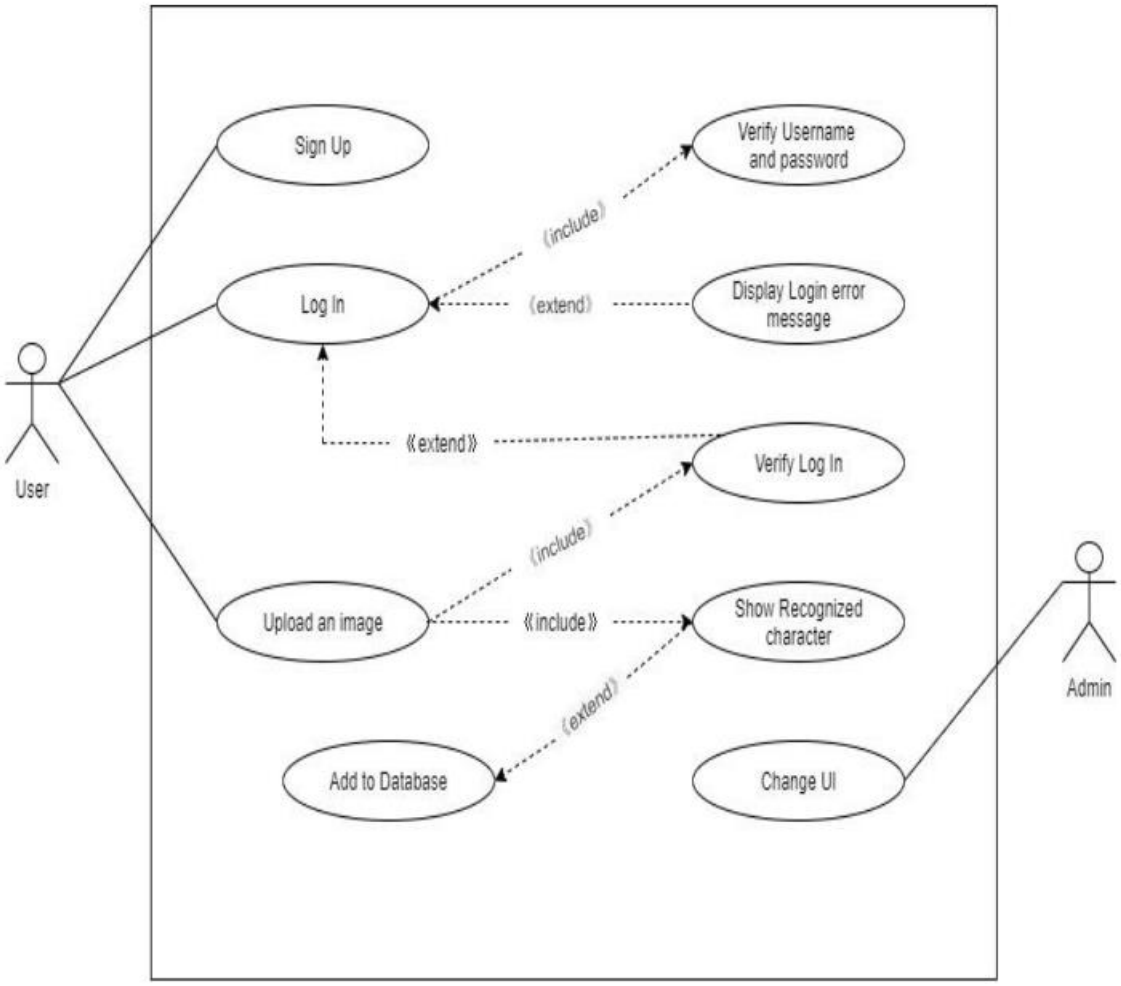


Figure 18- UseCase Diagram

Sequence Diagram

A sequence diagram shows how objects interact in a sequential order. It demonstrates how and in what order an object functions within a system. It's a common tool used by software developers and businesspeople in companies to text and realize requirements for new and accessible systems. Event diagrams and event scenarios are other names for sequence diagrams. They show how objects interact with one another in order to complete a task or a process. They are also meticulous planners who are well-versed in all aspects of current or upcoming condition. (Lucidchart, 2020)

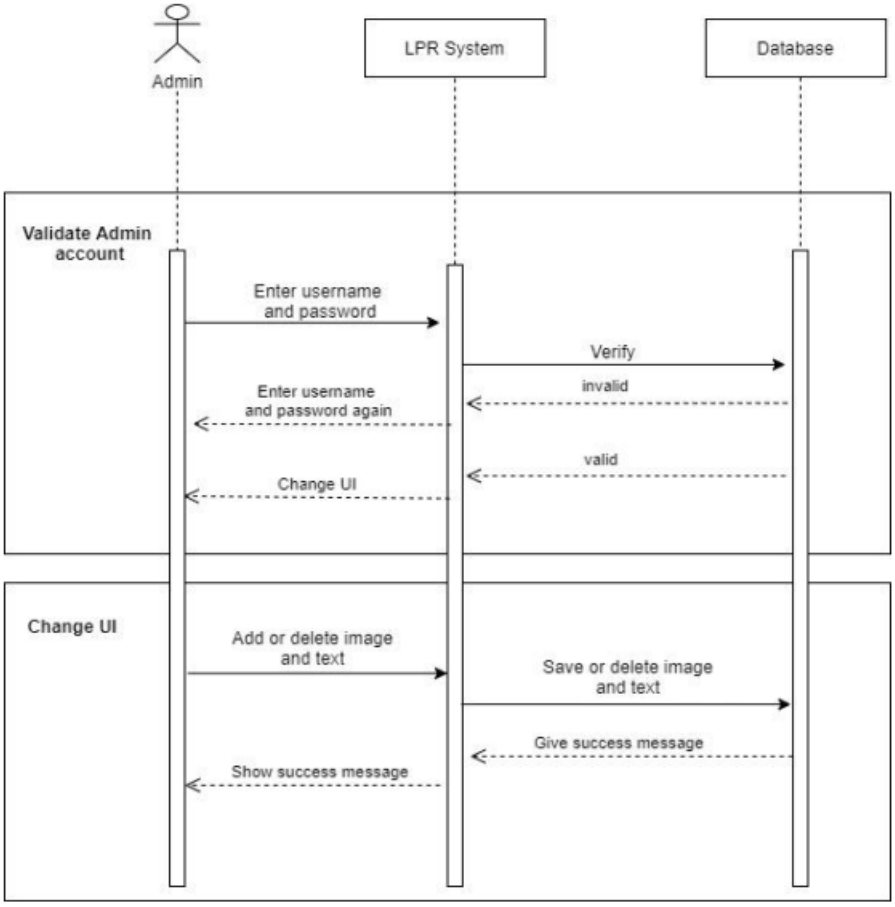


Figure 19-SequenceDiagram Admin View

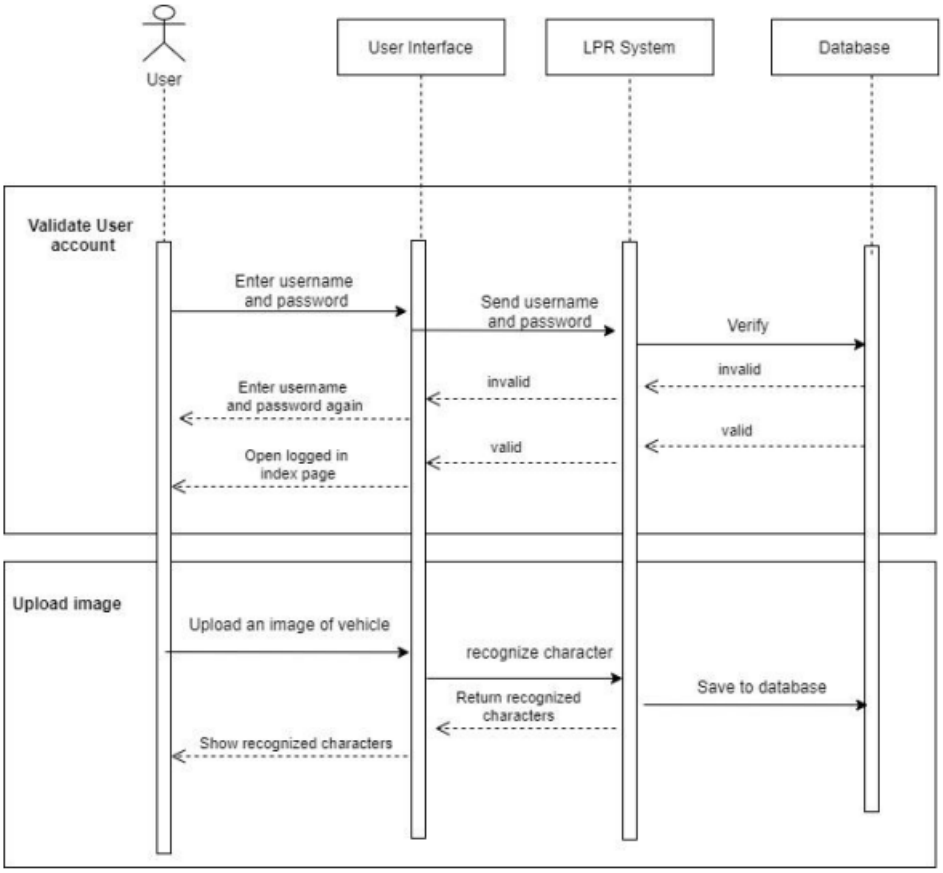


Figure 20-SequenceDiagram User View

Activity Diagram

The operation diagram, which is basically a flowchart or data flow diagram, is one of the most important diagrams in UML for showing the flow of control in the system. It shows how information flows from one operation to the next.

An activity diagram defines a degree of parallel and conditional tasks. It also goes into greater detail about use cases and system functions. With the help of an action sequence, it is used to create large concurrent activity flows. (techopedia, 2011)

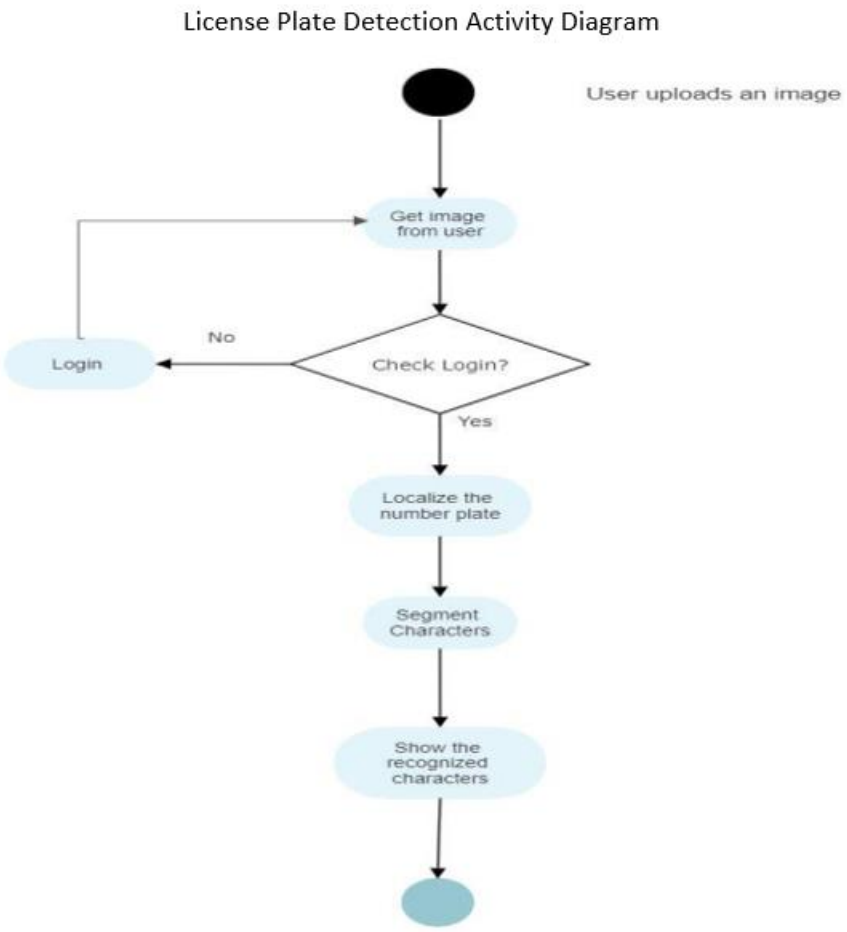


Figure 21-Activity Diagram

Class Diagram

A static diagrams depicting software's static view is known as a class diagram. It's a tool for visualizing, identifying, and recording the points of view of a system. It's also used to create software that can be executed. A class diagram is another name for a structural diagram.

The properties and processes of a class, as well as the system's constraints, are depicted in a class diagram. Since they can be directly mapped with object-oriented programming languages, class diagrams are widely used in the modeling of object-oriented systems.

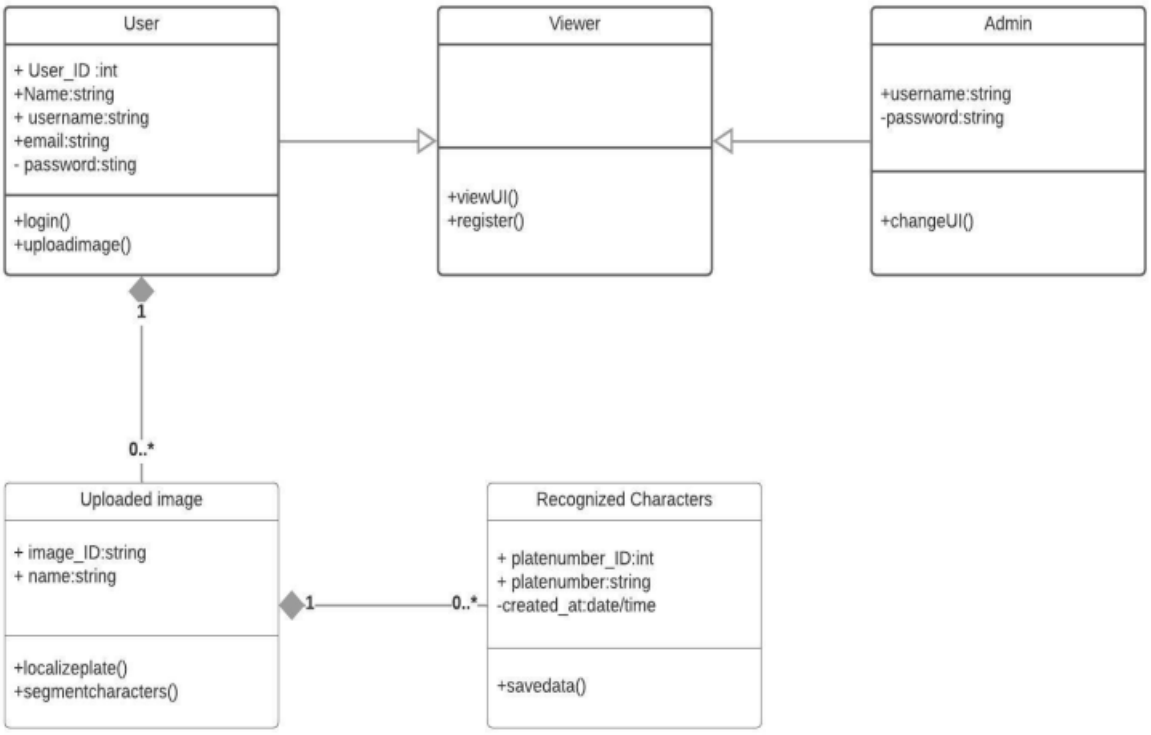


Figure 22- ClassDiagram

3.7 Database Design

Entity-relationship diagram (ERD)

An entity-relationship diagram depicts the relationship between the entity sets contained in a database (ERD). A piece of data known as an entity may be an individual or a variable. Entity attributes describe the properties of their corresponding entities. The conceptual structure of databases is depicted by an Entity-Relationship Diagram (ERD), which describes the entities, their attributes, and the relationships that exist between them. (SmartDraw, 2020) The database architecture for the proposed system is depicted in the entity-relationship diagram below.

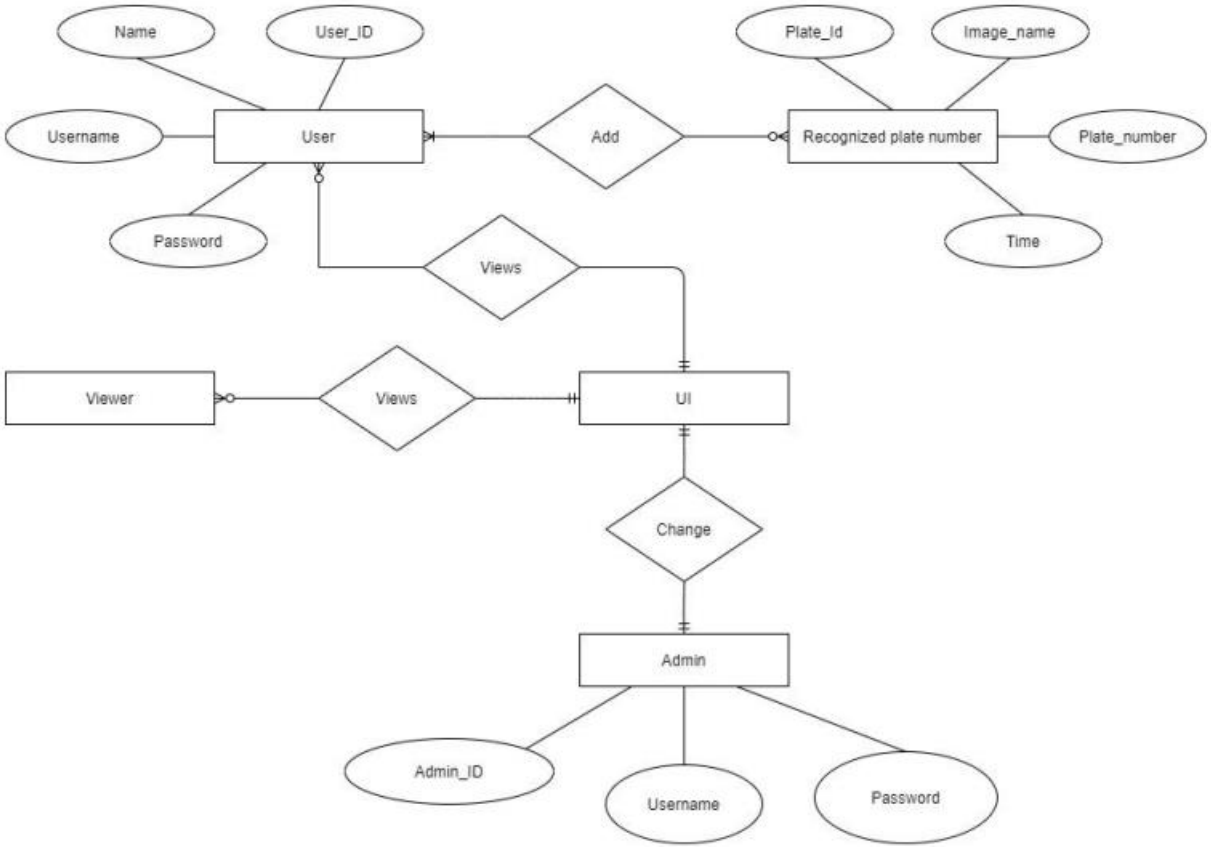


Figure 23-Entity Relation Diagram

Data Dictionary

External Entity	Field name	Data type	Key
User	User_ID	Int	Primary key
	Username	String	
	Name	String	
	Password	String	
Admin	Admin_ID	Int	Primary key
	Username	String	
	Password	String	
Platenumber	Platenumber_ID	Int	Primary key
	Platenumber	String	
	Date	Date/Time	

Figure 24- Data Dictionary

3.8 System Architecture

The design of a system's working structure is known as system architecture. Services, components, and their interactions are depicted in a system architecture diagram. The proposed system's architecture is represented below. . (J, 2018)

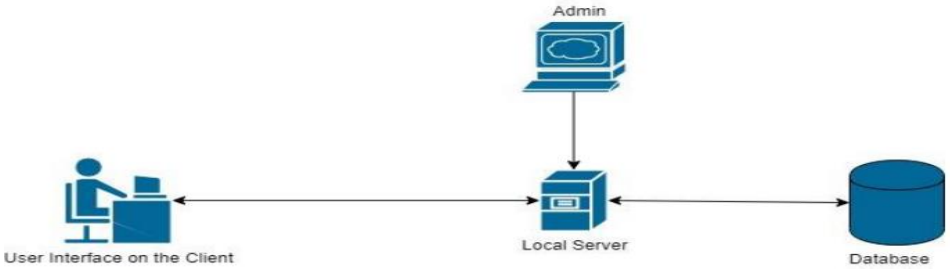


Figure 25-System architecture

Because it only runs on a local server, the proposed system has a simple system architecture. Connections to the User, Admin, and database are established by the

local server. The local server is used to transfer any data that needs to be transferred. As a result, if the local server goes down, the architecture's other nodes will be unable to complete their tasks.

4 Academic Question

Answer to the Academic Questions

- What is the best way to use CNN to recognize the characters?

I started with a dataset of 1408 training images and 412 testing images and built my model from the ground up. Data augmentation was done before using the CNN layers. The image's height and width were then entered as 32, 32 in the Convolutional layer. An image has three channels, as we all know: red, green, and blue. 3*3size kernels were defined for feature extraction.

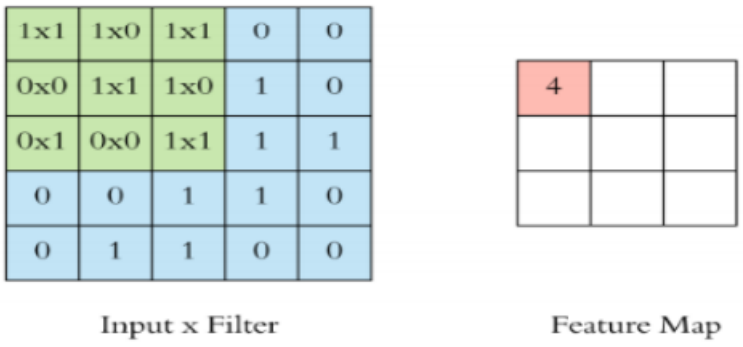


Figure 26- feature using convolutional layers

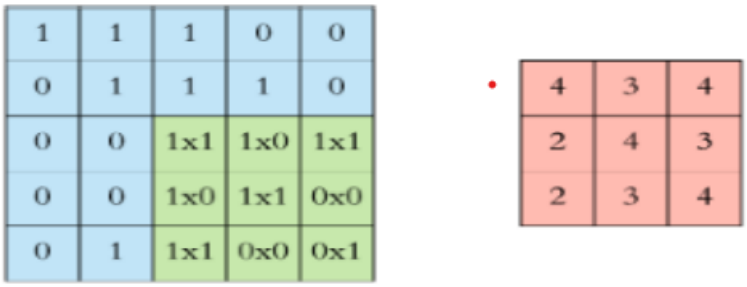
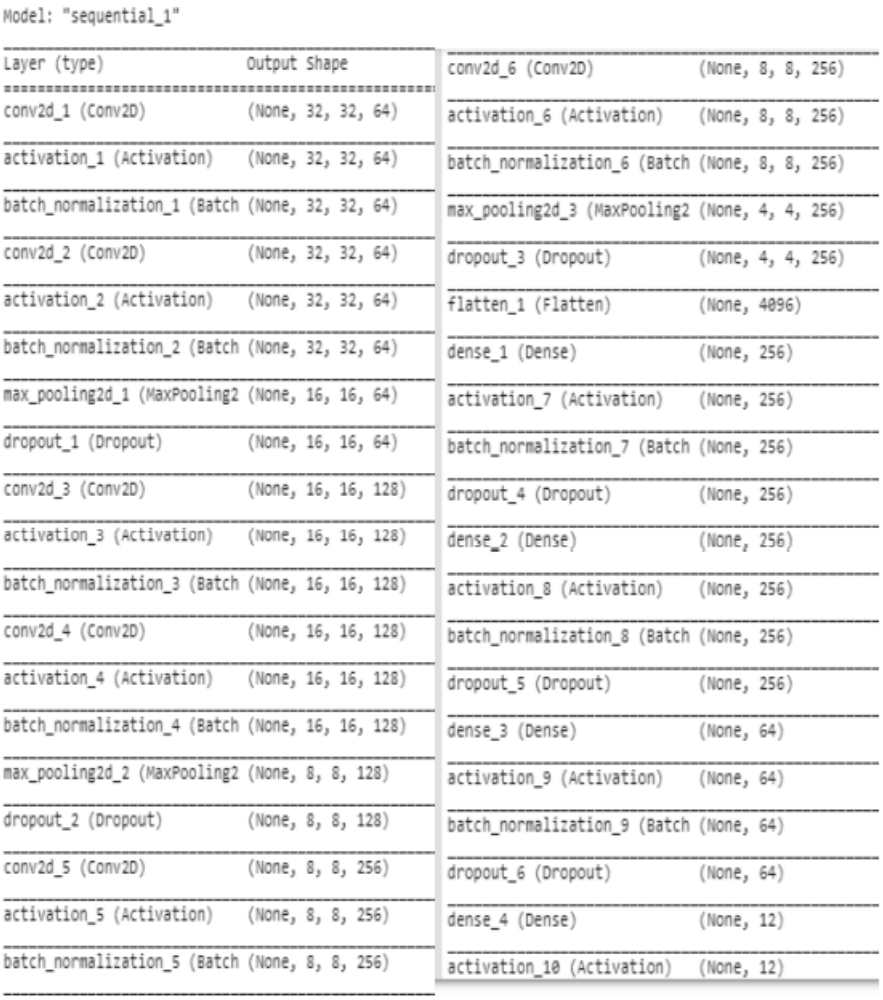


Figure 27- feature have been mapped

In the figures above, we can see how a function map is generated from an image using a 3*3 kernel. Function mapping is often performed in three dimensions since an image is represented as a three-dimensional matrix with dimensions of height, distance, and channels. To hold the function map the same size as the input image, padding was used. The ReLU function is added after the convolutional sheet, which replaces all negative values in the feature map with zero. The images were then normalized using Batch Normalization and MaxPooling2D with a scale of 44 2*2. We illustrate how MaxPooling works in the Literature Review segment. A dropout layer

with regularization of 0.2 and 0.5 was used to prevent the model from overfitting. Six sets of Convolutional layers were used, each with ReLU and BatchNormalization, and for every two sets of Convolutional layers, MaxPooling2D and Dropout layers were used.



```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              
=================================================================
conv2d_1 (Conv2D)            (None, 32, 32, 64)        
_________________________________________________________________
activation_1 (Activation)    (None, 32, 32, 64)        
_________________________________________________________________
batch_normalization_1 (Batch (None, 32, 32, 64)        
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 32, 32, 64)        
_________________________________________________________________
activation_2 (Activation)    (None, 32, 32, 64)        
_________________________________________________________________
batch_normalization_2 (Batch (None, 32, 32, 64)        
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 16, 16, 64)        
_________________________________________________________________
dropout_1 (Dropout)          (None, 16, 16, 64)        
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 16, 16, 128)       
_________________________________________________________________
activation_3 (Activation)    (None, 16, 16, 128)       
_________________________________________________________________
batch_normalization_3 (Batch (None, 16, 16, 128)       
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 16, 16, 128)       
_________________________________________________________________
activation_4 (Activation)    (None, 16, 16, 128)       
_________________________________________________________________
batch_normalization_4 (Batch (None, 16, 16, 128)       
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 8, 8, 128)         
_________________________________________________________________
dropout_2 (Dropout)          (None, 8, 8, 128)         
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 8, 8, 256)         
_________________________________________________________________
activation_5 (Activation)    (None, 8, 8, 256)         
_________________________________________________________________
batch_normalization_5 (Batch (None, 8, 8, 256)         
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 8, 8, 256)         
_________________________________________________________________
activation_6 (Activation)    (None, 8, 8, 256)         
_________________________________________________________________
batch_normalization_6 (Batch (None, 8, 8, 256)         
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 4, 4, 256)         
_________________________________________________________________
dropout_3 (Dropout)          (None, 4, 4, 256)         
_________________________________________________________________
flatten_1 (Flatten)          (None, 4096)              
_________________________________________________________________
dense_1 (Dense)              (None, 256)               
_________________________________________________________________
activation_7 (Activation)    (None, 256)               
_________________________________________________________________
batch_normalization_7 (Batch (None, 256)               
_________________________________________________________________
dropout_4 (Dropout)          (None, 256)               
_________________________________________________________________
dense_2 (Dense)              (None, 256)               
_________________________________________________________________
activation_8 (Activation)    (None, 256)               
_________________________________________________________________
batch_normalization_8 (Batch (None, 256)               
_________________________________________________________________
dropout_5 (Dropout)          (None, 256)               
_________________________________________________________________
dense_3 (Dense)              (None, 64)                
_________________________________________________________________
activation_9 (Activation)    (None, 64)                
_________________________________________________________________
batch_normalization_9 (Batch (None, 64)                
_________________________________________________________________
dropout_6 (Dropout)          (None, 64)                
_________________________________________________________________
dense_4 (Dense)              (None, 12)                
_________________________________________________________________
activation_10 (Activation)   (None, 12)                
```

Figure 28- The proposed system model

The layers were flattened into a 1D array, connected with a completely connected layer, and finished with a dense layer with the input as the number of classes using a softmax function after they were applied for feature extraction. Adam was used as an optimizer, and a loss function called "categorical cross-entropy for multi-class image classification" was used to create the model. 0.01 was chosen as the learning rate. To ensure that the model was properly fittedEarlyStop and ReduceLROnPlateau were used as callback tools. Callbacks were used to keep track of the validity failure.

The model will be stopped if the validity loss does not decrease after 5 epochs, and the best epoch with the least validation will be selected. Similarly, by multiplying the factor of 0.1 by 1, the ReduceLROnPlateau was used to reduce the learning rate. The model was then fitted and conditioned using the model.fit generator (). Each epoch had 704 training sample photographs, and the model was learned over 8 epochs.

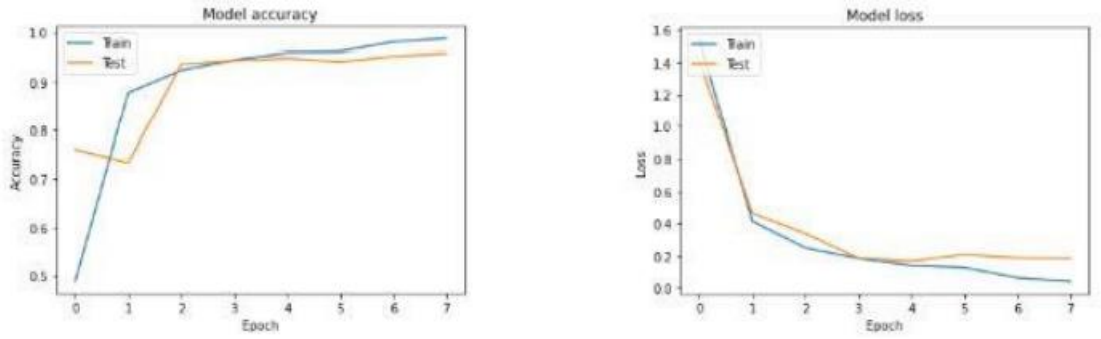


Figure 29- Accuracy and loss obtained in the model

Matplotlib was used to plot the graphs below after the model had been conditioned. In the most recent epoch, the preparation accuracy was 98.82 percent, and the validation accuracy was 95.88 percent. The model loss was reduced to 4% and the validation loss was reduced to 18% in the final epoch. Data augmentation is just as critical as feature extraction for this model's accuracy. As data augmentation attributes, I used rotation range, zoom range, distance shift range, and height shift range. Such characteristics were discovered to have a significant impact on character prediction accuracy.

```
train_datagen = ImageDataGenerator(
      rescale=1./255,
      rotation_range=10,
      zoom_range=0.1,
      width_shift_range=0.1,
      height_shift_range=0.1,
      horizontal_flip=True,
      fill_mode='nearest')
```

Figure 30- Image augmentation used in the model

- How accurate is CNN's recognition of the characters on the plate?

Following my model's training and the attainment of good accuracy and loss. To assess my model, I used a confusion matrix. In my first class's testing/validation set, I had 16 images, and in each of my remaining classes, I had 36 images.

```
Confusion Matrix
[[16  0  0  0  0  0  0  0  0  0  0  0]
 [ 0 36  0  0  0  0  0  0  0  0  0  0]
 [ 0  1 33  0  1  0  0  1  0  0  0  0]
 [ 0  0  0 33  0  1  2  0  0  0  0  0]
 [ 0  0  0  1 35  0  0  0  0  0  0  0]
 [ 0  0  0  0  0 35  0  0  0  1  0  0]
 [ 0  0  0  4  0  0 32  0  0  0  0  0]
 [ 0  0  0  0  0  0  0 36  0  0  0  0]
 [ 0  0  0  0  0  0  0  0 36  0  0  0]
 [ 0  0  0  1  2  0  0  0  1 32  0  0]
 [ 0  0  0  0  0  0  0  0  0  0 35  1]
 [ 0  0  0  0  0  0  0  0  0  0  0 36]]
```

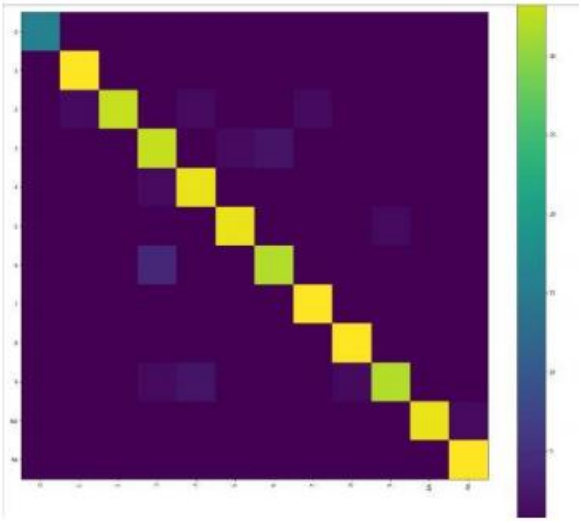


Figure 31- Graphical representation and Confusion matrix

	precision	recall	f1-score	support
0	1.00	1.00	1.00	16
1	0.97	1.00	0.99	36
2	1.00	0.92	0.96	36
3	0.85	0.92	0.88	36
4	0.92	0.97	0.95	36
5	0.97	0.97	0.97	36
6	0.94	0.89	0.91	36
7	0.97	1.00	0.99	36
8	0.97	1.00	0.99	36
9	0.97	0.89	0.93	36
BA	1.00	0.97	0.99	36
PA	0.97	1.00	0.99	36
accuracy			0.96	412
macro avg	0.96	0.96	0.96	412
weighted avg	0.96	0.96	0.96	412

Figure 32- Report of Classification

A classification report was generated after the confusion matrix was created, displaying the precision, recall, and f1-score for all available classes. For number recognition, the validation set yielded an overall accuracy of 94% . The Convolutional Neural Network could ensure that the characters from the license are well recognized after achieving such good results. I also used the CNN model to predict some sample images of separated characters to see how accurate it was at predicting the characters. This is evident in test case 5, which was successful in recognizing the characters.

5 WireFrames

5.1 Home Page

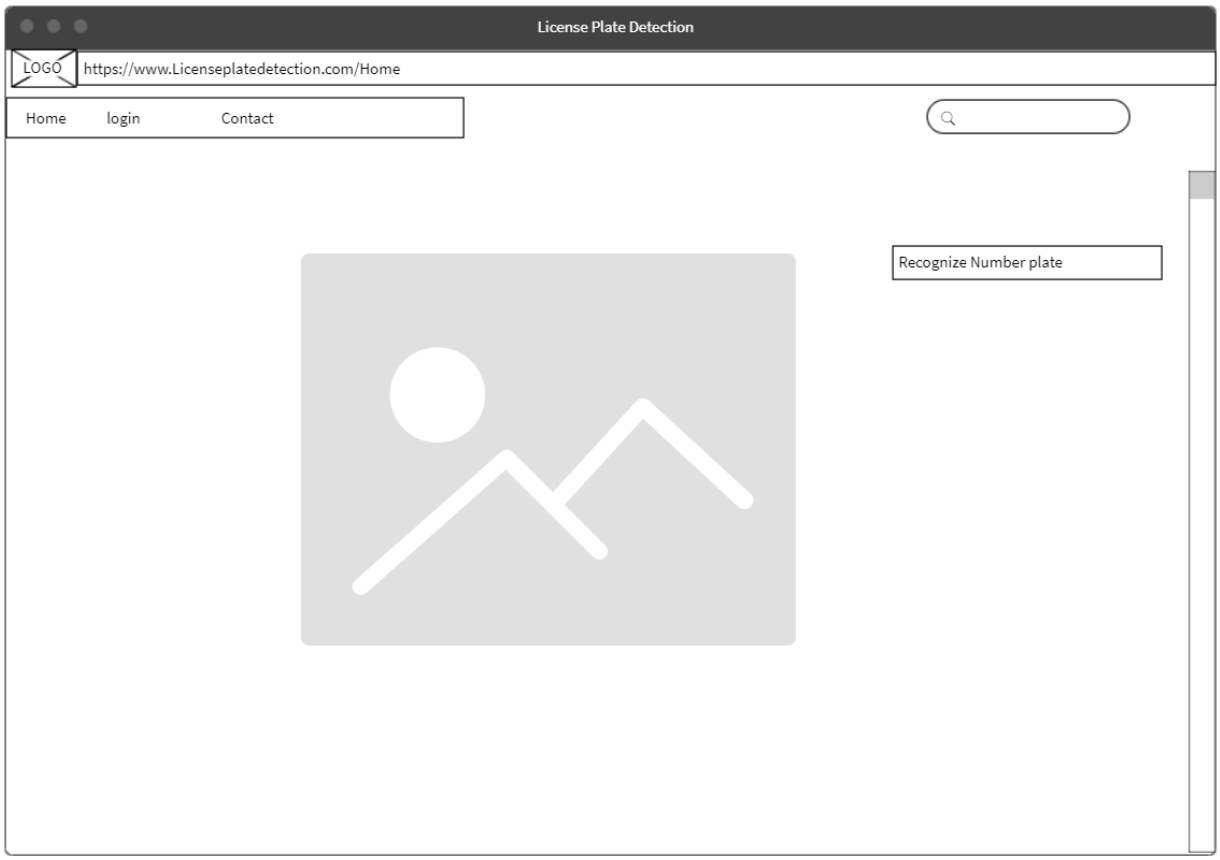


Figure 33-Homepage

5.2 Login Page

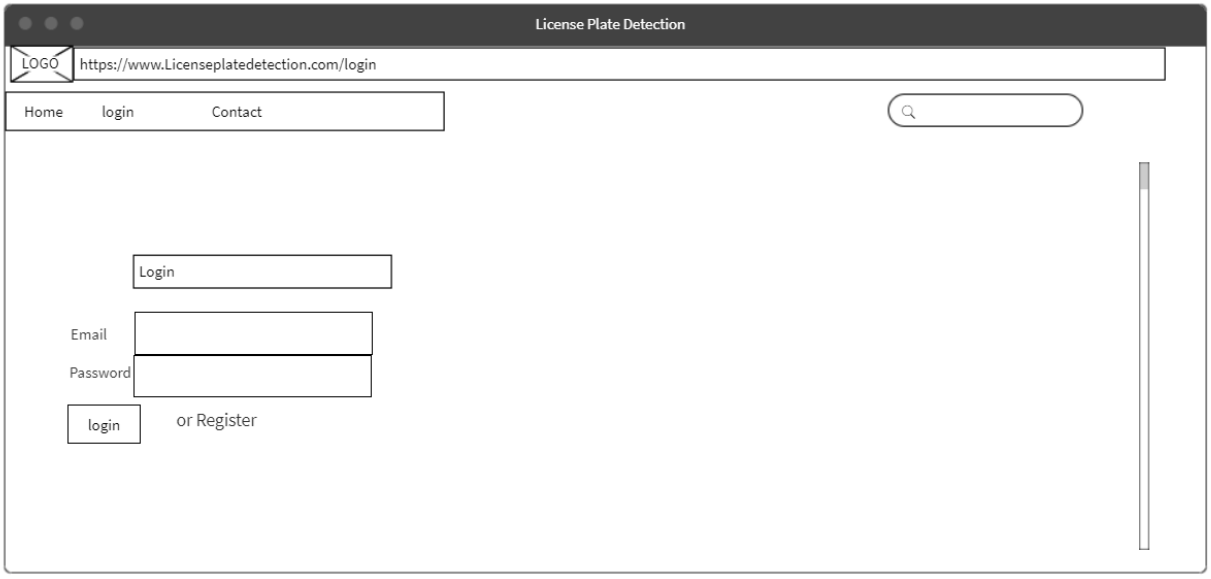


Figure 34- Loginpage

5.3 Signup Page

Figure 35-Signuppage

5.4 Contactus Page

Figure 36- ContactusPage

6 Testing

6.1 White Box testing

As various stages of assessment, the test cases were split into Unit testing and Integration testing and carried out using the white box testing process.

1. Unit Testing

Unit testing is a technique for determining whether or not individual units of source code are functional. That is, each unit of the program is tested independently in the developer's system.

Testcase1	Localization of license plate
Step1	Open web app and upload image
Expected Result	Image should be shown
Actual Result	Number plate was detected
Conclusion	Test was successful

Table 4-testcase 1

Figure 37-Testcase 1

Test Case 2	Saving the license plate
Step	Open web app and upload image
Expected Result	Number plate should crop and save
Actual Result	Number plate was display and saved
Conclusion	Test was successful

Table 5-testcase 2

Figure 38-testcase 2

Test Case 3	Segmentation of characters
Step	Open web app and upload image
Expected Result	The characters from the plate should be removed, and the names of segmented images should be saved in

	a segLists list, with the images saved in a folder.
Actual Result	Extract successfully
Conclusion	Test successfully

Table 6-testcase 3

```
print(segLists)

['roi0.jpg', 'roi1.jpg', 'roi2.jpg', 'roi3.jpg', 'roi4.jpg', 'roi5.jpg', 'roi6.jpg', 'roi7.jpg']
```

Figure 39-testcase 3

Test Case 4	Detecting contours and eliminating small contours
Step	Open web app and upload image
Expected Result	The contours were found and reduced.
Actual Result	As predicted, contours were reduced to ten.
Conclusion	Test successful.

Table 7-testcase 4

```
contours, hierarchy = cv2.findContours(final_image.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
print(len(contours))

contours = sorted(contours, key=cv2.contourArea, reverse=True)[:10]

print(len(contours))

621
10
```

Figure 40-testcase 4

TestCase 5	Sample characters prediction
Step	For sample prediction, four images from each class were separated from the dataset. Following the development of the artificial intelligence model, samples were tested.
Expected Result	The original class to which the image belongs is Orig, and the predicted class is Pred. The expectation was that the majority of the images would be correctly classified.
Actual Test Result	The majority of images in the sample prediction were correctly classified, as can be seen.
Conclusion	Test Successfully

Table 8-testcase 5

Figure 41-testcase 5

2. Integration Testing

The process of evaluating the compatibility or data transmission between a few of the unit's modules is known as integration testing.

Test Case 6	Recognition of characters
Step	Open web app and upload image
Expected	The characters on the plate should be easy to identify.
Actual Result	The characters was recognized.
Conclusion	Test was successful.

Table 9-testcase6

Figure 42-testcase6

6.2 Black Box Testing

Black Box Testing areas:

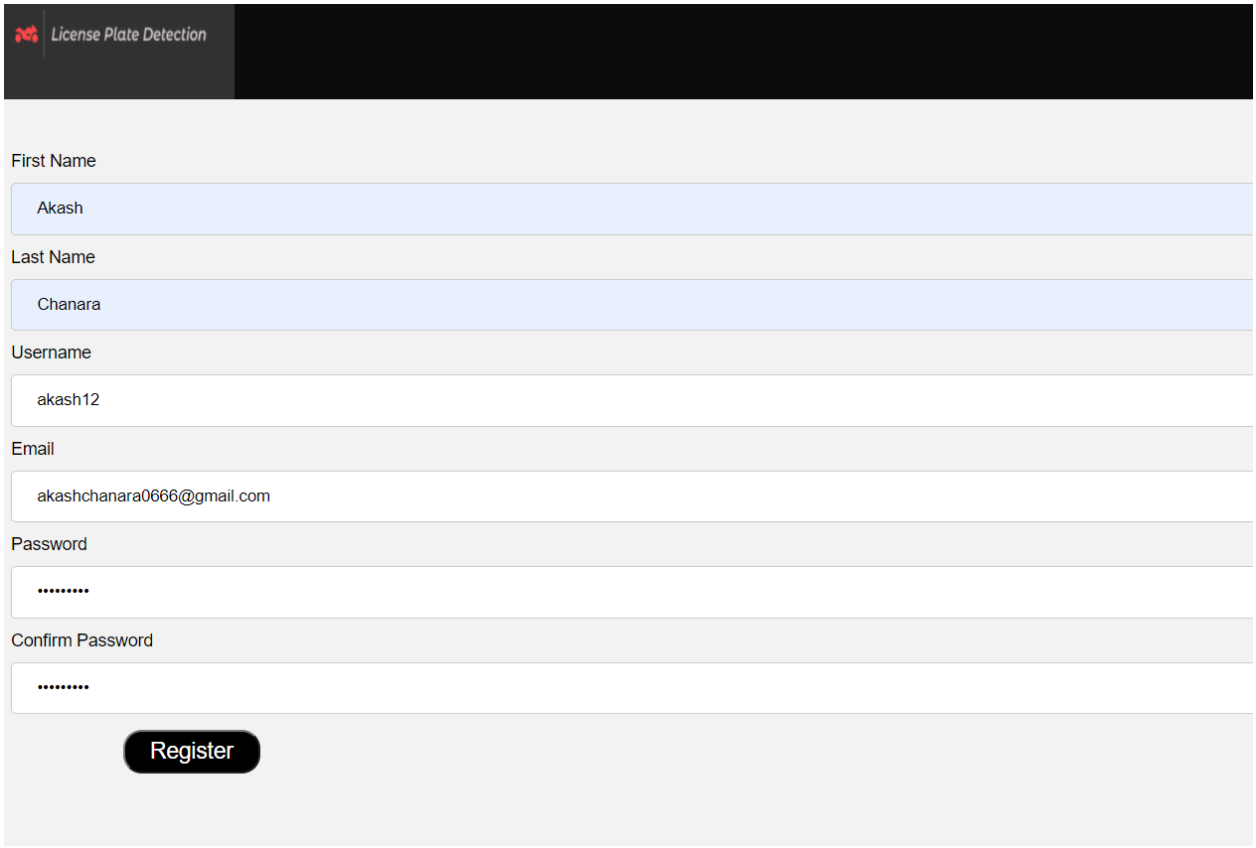
TestCase 1	Login
Step1	Open the License Plate Detection application
Step2	Login to account, You need to register if you do not have an account
Expected Result	Your name should be shown in the navigator bar after you click the login button
Actual Result	Name as shown in the navigator bar.
Conclusion	The test was successful.

Table 10-TestCase 1

Figure 43-TestCase 1

Test case 2	Register
Step 1	Input valid user name ,email
Step 2	Password must match
Expected Result	User Created
Actual Result	User Created
Conclusion	Test was successful

Table 11-TestCase 2



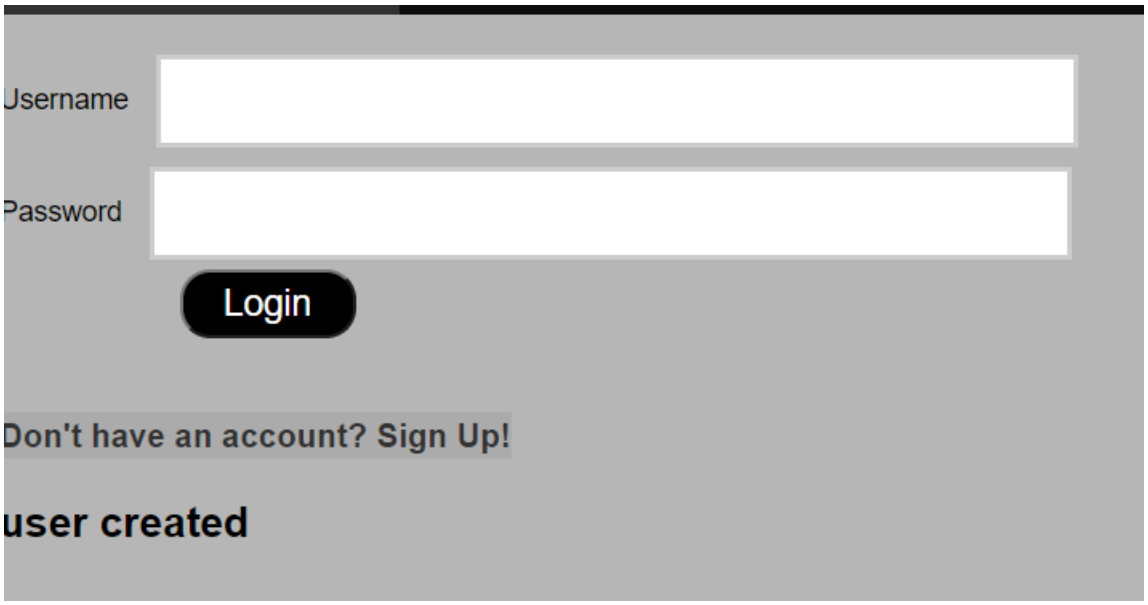


Figure 44-TestCase 2

Test case 3	Register
Step 1	Open web application and login to web app
Step 2	Input image
Expected Result	License number should be display
Actual Result	License number is displayed
Conclusion	Test was successful

Table 12-testcase3

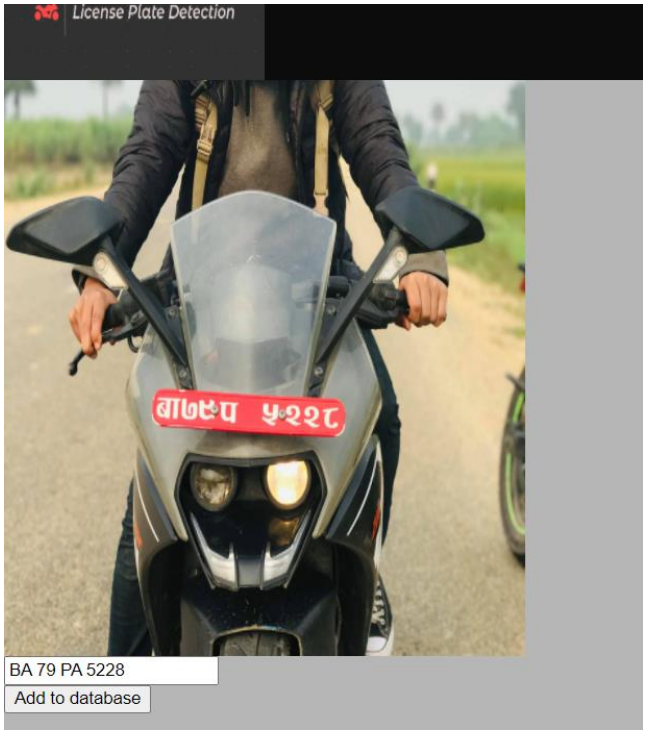


Figure 45-testcase3

6.3 Gannt Chart

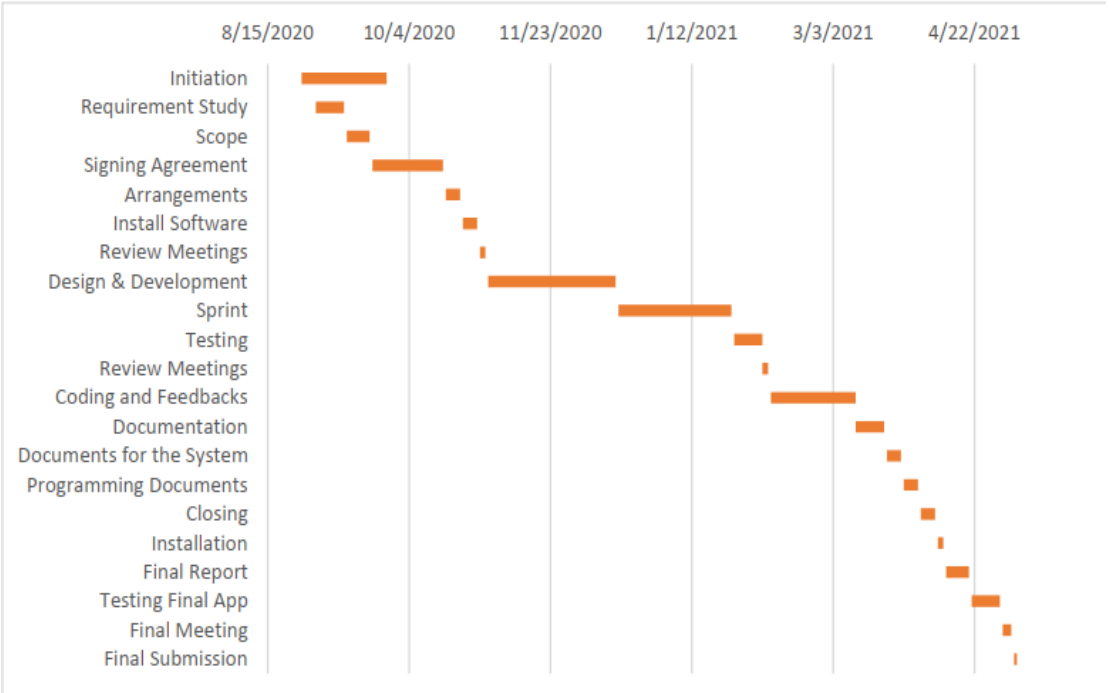


Figure 46- Gantt Chart

7 Conclusion

Image processing is a difficult task, as I learned from this project. Many people around the world are still attempting to figure out how to accomplish this. One of the most difficult tasks of image processing is recognizing license plate numbers. I learned various image preprocessing techniques used in various types of images while working on this project. Removing noise from an image before processing improves image processing accuracy significantly. Then, while creating a Convolutional Neural Network model for use in my system, I learned parameter tuning, which helped to improve the model's accuracy.

The proposed system performed character recognition, but due to a lack of dataset, it could only recognize the character of a private vehicle in the Bagmati zone. Keras libraries and a Convolutional Neural Network were used to train the dataset. The contours detection method was used to localize the plate, and connected component analysis and contours detection were used to extract the characters.

Expanding the dataset to include the All vehicles registered in all zones of Nepal have unique characters that can assist the device in recognizing them and improve the character of all vehicles in Nepal. For the localization of license plates of various colors, I discovered that different masks were required in this project. Different masks should be built for better plate extraction because Nepali license plates come in a variety of colors for private, public, organizational, and governmental vehicles. I was able to locate the plate in the majority of the images I tested. Character extraction was unsuccessful because the sizes of numbers obtained from zoomed in and zoomed out images were not well matched. As a result, the system failed to recognize the characters, and the segmented characters did not match the license plate in order. The characters from the plate could be effectively segmented using a vertical and horizontal projectile method based on the Peak and Valley concept.

8 Critical Evaluation

After the system development was completed, the report's Main Content was started so that each device function could be reflected in the section after that In the Academic Questions segment, the question from the introduction was briefly answered. Different knowledge was acquired in the process of researching different papers and similar applications, which was given more time.

Different algorithms that can classify the image into their respective groups are discovered after the model has been trained. Large computations are needed for the model's training, which can take a long time on the CPU. The training of the model could be faster if it is done on a GPU-enabled device. It also depends on how much data we're training with. If the amount of data to be trained is large, training the model without the use of GPU devices can be nearly impossible.

The model was trained using Google Colaboratory, an online platform that lets us write and run Python in our browsers while also giving us free access to a limited amount of GPU support. Because the dataset containing the images I used was small, I was able to train the model to a satisfactory level of accuracy in a short amount of time using the GPU provided by Google Colaboratory. Pickle was used to save the developed model in.pkl format for easy serialization and deserialization while integrating with the User Interface.

9 References

Abu, M. A. e. a., 2019. International Journal of Engineering Research and Technology. *A study on Image Classification based on Deep Learning and Tensorflow,* Volume 12, pp. 563-569.

Ashok Kumar Pant, P. K. G. &. S. A., 2015. International Conference on Software, Knowledge, Information Management and Applications (SKIMA). *Automatic Nepali Number Plate Recognition with.*

Bharat, B. S. S. &. S. R., 2013. International Journal of Computer Applications. *License License,* pp. 45-50.

Brownlee, J., 2019. *machinelearningmastery.* [Online]
Available at: https://machinelearningmastery.com/loss-and-loss-functions-for-training-deep-learning-neural-networks/
[Accessed 03 04 2021].

Cavus, N. &. T. S. S., 2016. 12th International Conference on Application of Fizzy Systems and Soft Computing. . *A Hybrid KNN-SVM Model for Iranian License Plate Recognition,* pp. 1-7.

Chigozie Enyinna Nwankpa, W. I. A. G. a. S. M., 2018. *Activation Functions: Comparison of Trends in Practice and Research for Deep Learning.*

Daniel, G., 2018. *Understanding binary cross-entropy / log loss: a visual explanation,* s.l.: s.n.

Forest Agostinelli, M. H. P. S. P. B., 2015. *Learning Activation Functions to Improve Deep Neural Networks,* s.l.: s.n.

Geoffrey Hinton, N. S. K. I. S. a. R. S., 2014. Journal of Machine Learning Research. *Dropout: A Simple Way to Prevent Neural Networks from,* pp. 1929-1958.

Guru99, 2020. *Guru99.* [Online]
[Accessed 19 3 2021].

Himanshu Jain, Y. P. V., 2016. *Extreme Multi-label Loss Functions for Recommendation, Tagging, Ranking & Other Missing Label Applications,* s.l.: s.n.

iphone.apkpure, 2020. *iphone.apkpure.com.* [Online]
Available at: https://iphone.apkpure.com/maxsoft-mobile-lpr/com.maxsoft.mobilelpr

J, S., 2018. *Simplicable.* [Online]
Available at: https://simplicable.com/new/system-architecture-definition
[Accessed 12 03 2021].

Kenton, W., 2019. *Investopedia.* [Online]
Available at: : https://www.investopedia.com/terms/f/functional-decomposition.asp
[Accessed 13 2 2021].

Kocer, H. E. &. C. K. K., 2010. Procedia Computer Science.. *Artificial neural networks based vehicle license plate recognition,* pp. 1033-1037.

Li-Shien Chen, S.-L. C. Y.-C. C. a. S.-W. C., 2004. IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS. *Automatic License Plate Recognition,* Volume 5, pp. 42-53.

logo, V. P., 2020. *Visual Paradigm logo.* [Online]
Available at: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/
[Accessed 20 03 2021].

Lucidchart, 2020. *Lucidchart.* [Online]
Available at: https://www.lucidchart.com/pages/uml-sequence-diagram
[Accessed 21 03 2021].

Meghana, P. S. S. S. P. &. S. K., 2019. International Journal of Innovative Technology and Exploring Engineering (IJITEE).. *Image Recognition for Automatic ,* 8(4), pp. 9-12.

Mordvintsev, A. &. R. A. K., 2013. *Opencv.* [Online]
Available at: https://opencv-pythontutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_thresholding/py_thresholding.html
[Accessed 14 10 2020].

O. R. Vincent, O. F., 2009. *A Descriptive Algorithm for Sobel Image Edge Detection,* s.l.: Proceedings of Informing Science & IT Education Conference (InSITE).

OpenCV, 2020. *openCv.* [Online]
Available at: https://docs.opencv.org/3.4/d4/d73/tutorial_py_contours_begin.html
[Accessed 13 10 2020].

P. Surekha, P. G. R. P. a. V. R. A., 2018. ICTACT JOURNAL ON IMAGE AND VIDEO PROCESSING. *AUTOMATIC LICENSE PLATE RECOGNITION USING IMAGE PROCESSING AND NEURAL NETWORK,* pp. 1786-1792..

Patel, C. S. D. &. P. A., 2013. 2013. International Journal of Computer Applications. *Automatic Number Plate Recognition System (ANPR): A Survey ,* 69(9), pp. 21-33.

platerecognizer, 2020. *platerecognizer.com.* [Online]
Available at: https://platerecognizer.com/?utm_source=google&utm_medium=cpc&utm_campaign=72300747320&utm_keyword=recognition%20plate&utm_content=1011034&gclid=CjwKCAjww5r8BRB6EiwArcckC7yfgEq333xlZMbdA8BtdYrZEJP3kmnWTYP5OBVRH1dNmn9EE8nhERoCTpIQAvD_BwE

Radhika Chandwadkar, S. D. V. R. P. S. A. T., 2013. *Comparison of Edge Detection Techniques.*

Ravi S, A. M. K., 2013. National Conference on VLSI, Signal processing & Communications. *Srisha, R. & Khan, A., 2013. Morphological Operations for Image Processing : Understanding.*

Saad ALBAWI, T. A. M. A.-Z., 2017. *Understanding of a Convolutional Neural Network,* Turkey: ICET.

Saha, S., 2018. *towardsdatascience.* [Online]
Available at: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53
[Accessed 1 04 2021].

Samer Hijazi, R. K. a. C. R. I. G. C., 2015. *Using Convolutional Neural Networks for Image Recognition,* s.l.: s.n.

SHARMA, S., 2017. *towardsdatascience.* [Online]
Available at: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6
[Accessed 04 03 2021].

SmartDraw, 2020. *SmartDraw.* [Online]
Available at: https://www.smartdraw.com/activity-diagram/
[Accessed 11 03 2021].

SmartDraw, 2020. *SmartDraw.* [Online]
Available at: https://www.smartdraw.com/entity-relationship-diagram/
[Accessed 16 03 2021].

Szandała, T., 2017. *Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks,* s.l.: s.n.

techopedia, 2011. *techopedia.* [Online]
Available at: https://www.techopedia.com/definition/27489/activity-diagram
[Accessed 12 03 2021].

Times, T. H., 2020. *Traffic cops to treat hit-and-run cases more seriously,* kathmandu: s.n.

Togashi, R. Y. &. N. K. G. D., 2018. *Convolutional neural networks: an overviewand application in radiology,* pp. 611-629.

towardsdatascience, 2020. *towardsdatascience.com.* [Online]
Available at: https://towardsdatascience.com/batch-normalization-in-neural-networks-1ac91516821c
[Accessed 14 10 2020].

Uniqtech, 2018. *medium.com.* [Online]
Available at: https://medium.com/data-science-bootcamp/understand-the-softmax-function-in-minutes-f3a59641e86d
[Accessed 01 03 2021].

Zenghui, W. R. W., 2017. *Deep Convolutional Neural Networks for Image Classification,* Volume 29, pp. 2352-2449.

Appendix

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting
Proposal writing.

What do you aim to complete before the next meeting
Research about license plate detection system

Supervisor comments
Research about license plate detection system

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date: 10/9/2020

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	**Month: September**

What have you done since the last meeting

Proposal writing.

What do you aim to complete before the next meeting

Start working on the project and build a workable sub-system

Supervisor comments

Few changes required for report otherwise all the things is good.
Correct aims and objectives

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: Sign Date: 13/9/2020

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	**Month: September**

What have you done since the last meeting

Improved mt proposal as per feedback given by supervisor.

What do you aim to complete before the next meeting

Create UI

Supervisor comments

Create login/register system

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date: 21/9/2020

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Literature review writing.

What do you aim to complete before the next meeting

Create login/register system

Supervisor comments

Complete writing Literature review

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date: 28/9/2020

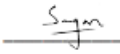
Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting
Create login/register page.
What do you aim to complete before the next meeting
Complete literature review part of my project
Supervisor comments
Complete UI and train module

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: 

Date:05/10/2020

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Improved my report as per feedback given by supervisor

What do you aim to complete before the next meeting

Train module with new image

Supervisor comments

- Train module with new image .
- Put Algorithm Explanation inside Literature review and have it formatted in a better way so that it is easily readable . May be show how it works with images or fig and explained it .

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: Sagar

Date:12/10/2020

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Train module with new image

What do you aim to complete before the next meeting

Start writing Artefact

Supervisor comments

Complete writing artefact and check dataset .
Add wireframe and test case.

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature:  Date:30/11/2020

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Done Artefact design in part of my report

What do you aim to complete before the next meeting

Complete writing professionalism report.

Supervisor comments

Start writing professionalism report .
Functional requirement should be in a format,

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date:7/12/2021

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting
Complete writing professionalism report

What do you aim to complete before the next meeting
Learn about YOLO

Supervisor comments
Learn about YOLO and data gathering

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: 

Date:14/12/2022

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Complete writing professionalism report

What do you aim to complete before the next meeting

Integrate Artificial Intelligence part in web Application

Supervisor comments

Explain how it will be implemented in your system.

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date:4/01/2021

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Complete writing professionalism report

What do you aim to complete before the next meeting

Train YOLO module and try to open webcam

Supervisor comments

Install a drive to open webcam and solve a error

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date:11/1/2021

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Open webcam from visual studios

What do you aim to complete before the next meeting

Open webcam and capture image of license plate

Supervisor comments

Try to use YOLO and capture image of license plate

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: Sign Date:25/1/2021

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Open webcam from visual studios

What do you aim to complete before the next meeting

Create a sub section for algorithm.

Supervisor comments

Show the clearer working mechanism in your system.

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date:1/3/2021

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Completed the Artificial Intelligence part of my project

What do you aim to complete before the next meeting

Integrate Artificial Intelligence part in web application and merger all the draft report.

Supervisor comments

Integrate artificial intelligence part in web application.

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date:4/3/2021

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG	
First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Completed the preprocessing part of datasets

What do you aim to complete before the next meeting

I am planning to complete detection part.

Supervisor comments

Complete the detection part
Merge all the draft of report in one report.

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature: _____ Date:15/3/2021

Faculty of Science and Engineering
School of Mathematics and Computer Science

UNIVERSITY OF WOLVERHAMPTON

PROJECT MANAGEMENT LOG

First Name: Akash	**Surname: Chanara**
Student Number:	**Supervisor: Mr. Sagar Lamichhane**
Project Title: License Plate Detection	

What have you done since the last meeting

Merge all the draft report in a final report.

What do you aim to complete before the next meeting

Cover all the topic mention in the final year project templates.

Supervisor comments

Check the font size and line spacing
Change the numbering format

We confirm that the information given in this form is true, complete and accurate.

Supervisor Signature:  Date:26/4/2021