

Effective Plate Object Segmentation and Recognition by Implementing Artificial Intelligent Model for Smart Transportation Systems.

^[1] Mr. Mehul T. Patel, ^[2] Dr. Ashishkumar Parejiya

^[1] Ph. D. Research Scholar – Indus University, Rancharda, Via, Thaltej, Ahmedabad, Gujarat – 382115

^[2] Ph.D.- Computer Science Research Supervisor– Indus University & Director Technology, Piramal Group of Industries, Mumbai, Maharashtra - 400070.

E-mail: ^[1] mehulpatel.rs@indusuni.ac.in, ^[2] drashishparejiya@gmail.com

Abstract: The increase in the number of vehicles in the last few years has made it challenging to manually note the number plate text of the vehicle. Hence, in order to reduce the manual work, there is a need to propose a methodology that can detect the number plate region from the input image and recognize the characters of the number plate. Systems have been built for the same using Image Processing techniques, but this technique fails to provide accurate results occasionally in the case of real data. Modern technology such as Deep Learning overcomes this problem. Hence, a deep learning-based methodology is proposed to detect the number plate region from the input image and recognize its characters. Using the pre-trained based deep learning Networks, the number plate region is detected and using Convolutional Neural Networks (CNN), the characters are recognized from the detected plate region. The system also stores the number plate text with its state name into the database to maintain a record of number plates detected. The proposed system provides promising results. The model is judged against other models that are already out there, such as ResNet50, DenseNet, and DenseCapsNet. According to these studies, MobileNet has reached a level of accuracy of 98.69%, which is higher than other algorithms that are thought to be cutting-edge.

1. Introduction

Due to the increasing number of vehicles nowadays, the modern city needs to establish the effective and efficient automatic traffic system for the management of the traffic law enforcement. Number plate recognition leads the significant role in this condition. The number plate recognition is an image processing technique to extract the image of license plate on vehicle taken by digital camera or taken by a color or a grayscale digital camera, as well as an infrared camera in order to identify the vehicles using their number plate. The Number Plate Recognition system recognizes characters on license plate through the combination of various techniques and algorithms, including image pre-processing, object detection, character segmentation and recognition. It consists of a camera to detect the number plate object and processing unit to process and extract the characters and interpret the pixels into numerically readable characters.

The ANPR system has been used in traffic law enforcement, including speed camera, traffic light camera, stolen car detection, and border monitoring. It can be used also for building management, such as parking management and gate control. With the rapid development of highways and the wide use of vehicles, people have started to pay more and more attention to advanced, efficient, and accurate Intelligent Transportation Systems (ITSs). The Number plate recognition task is quite challenging from vehicle images due to the viewpoint changes when vehicle bodies and license plates have similar colors, multi-style plate formats, and the non-uniform outdoor illumination conditions during image acquisition. Machine learning approaches the problem in a different way. The idea is to take a large number of number plates, known as training data, and then develop a system that can learn from those training examples. In other words, machine learning uses the examples to automatically infer rules for recognizing number plates. Furthermore, by increasing the number of training examples, the network can learn more about numbers and characters, and so improve its accuracy. Due to the increasing number of vehicles, manually controlling and monitoring traffic is time-consuming, costly, inaccurate, and sometimes impossible. This makes automatic vehicle plate recognition a recurrent research topic. Since the plate is the unique ID of vehicles, several prominent applications are found for automatic plate recognition, including traffic control,

driving offense detection, vehicle speed estimation, self-driving vehicles, and surveillance [1]. To this end, many cameras are installed in cities, roads, highways, borders, parking lots, and protected areas for better and more accurate control of vehicles. These cameras are constantly monitoring the images of passing vehicles. Vehicles and their plates cannot be detected and recognized without processing and analyzing these images. There is a need for a system based on image processing and machine learning to detect vehicles and extract other information such as plate numbers [2]. In vehicle plate detection and recognition systems, the quality of the input images has a direct impact on the result accuracy and processing speed. Many parameters are influential in the quality of the captured images including environmental and weather conditions such as light projection angle, light intensity, rainfall, fog, dust, humidity, dark, glare, occluded, rainy, snowy, tilted, or blurred scenarios. There are other concerns in automatic plate recognition systems, including different plate alignments on vehicles, size, plate aspect ratio, various shapes of plates, various angles of camera placement, too low or too high lighting, presence of several plates in the image, various plate background colors, excessive plate dirtiness, and various arrangements of letters and numbers in plates [3, 4]. As shown in Figure 1, a vehicle plate detection and recognition system consists of the three main steps: (i) plate detection, (ii) segmentation of characters in the plate, and (iii) character recognition [5]. The most important and challenging step is plate detection. If plates are not detected properly, the subsequent steps will not work as expected and the final result will be entirely wrong.

Not only can modern ANPR cameras read plates, but they can also provide useful additional information such as counting, direction, groups of vehicles, and their speed. The ability to detect and read large volumes of fast-moving vehicles has meant that ANPR technology has found its way into many aspects of today's digital landscape. Whilst ANPR technology can come in many different packages, they all perform the same basic function which is to provide a highly accurate system of reading a vehicle without human intervention. It is utilized in very diverse applications such as access control, parking management, tolling, user billing, delivery tracking, traffic management, policing and security services, customer services and directions, the red light and lane enforcement, queue length estimation, and many other services [2–8]. Figure 1 shows the basic system diagram of a fixed and mobile ANPR technology.

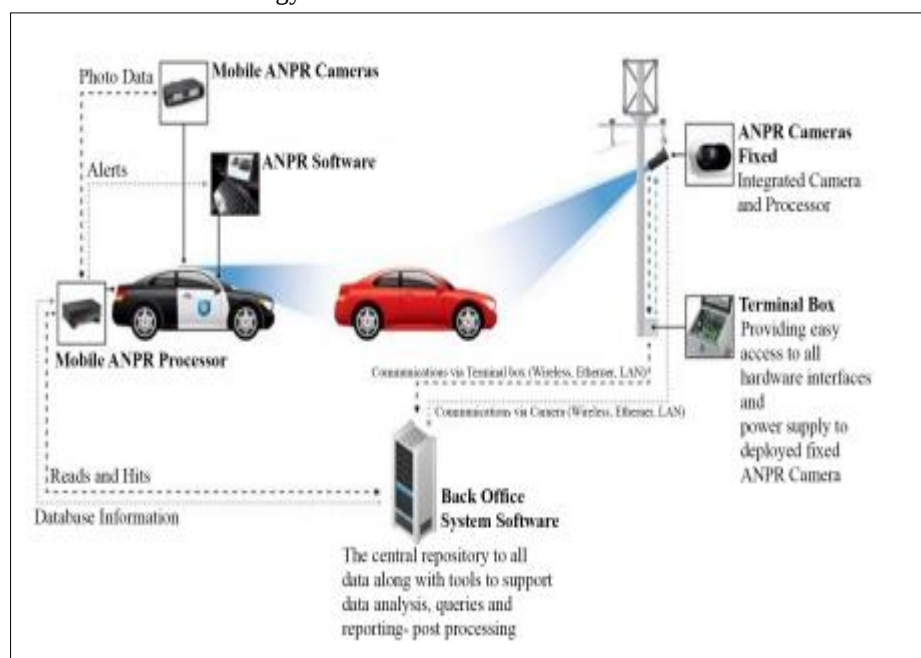


Fig 1: Typical ANPR System Diagram of a Fixed ANPR System (right) and a Mobile ANPR System (left)

2. Literature Review

Some protocols developed previously will be discussed in this section. A significant amount of work has been done over the last couple of years on image processing techniques and deep learning for object detection purposes. Several different recognition and detection algorithms for vehicle reconnaissance have evolved in this field. We can see different current techniques occurring from the literature review. K.K. Kim built a license plate recognition system by following a learning protocol. The camera captures an image inside the car detection

module. Then, the picture of the candidate region is provided as output. The two TDNNs were taken as the horizontal and vertical filters to find the license plate. The segmentation rate was 97.5 percent, and a recognition rate of 97.2 percent for the proposed system [1].

Chin-Chuan Han suggested a device that not only tracks several targets but also obtains high-quality images on plate numbers. A computer with a tuned dual-camera system has been built here by the author; a stationary camera and a pan-tilt-zoom camera are designed to monitor moving conveyance in an open field. The license plate for recognition has been sequentially identified by CNN classifier. As 64 vehicles entered this region illegally, data was composed manually from the science pictures, and 59 IDs were accurately detected using this tool [2].

The identification of license plates is a very good source of knowledge for record detection and recognition. However, the conventional license plate recognition process is a boring one. The manual way of identifying the vehicle and its owner is not that applicable in detecting license plates to retrieve hidden treasures of information. Automatic recognition of license plates is an important stage in the intelligent traffic network, and several ways for the construction of ANPR architecture have been developed. Although the proportion of the license plate in the image is greatly correlated with shooting distance, therefore the ANPR architecture is not easy to balance. Vehicles in motion, however, are too tiny to capture in the huge open space and clear recognizable license plate images. Identifying a license plate from tiny and distorted pictures will reveal a lot of effort. One answer is to use a CCD camera with panning, tilting, and zooming (PTZ) capturing functions. The various researchers have proposed various techniques for every step and an individual technique has its own pros and cons. The method for recognizing license plate includes the three main steps. That is the region of extraction of interest, extraction of plate numbers and recognition of character. Below is the block diagram for the license plate system. License plate recognition is achieved through device program ANPR. Images taken by using cameras displays the output of vehicle plate. ANPR's basic work is reading and unfolding license plate.

Madhusree Mondal in 2017 developed an ANPR framework focused on the learning capabilities of convolutional neural networks. The self-synthesized function of CNN was used here, as it distinguishes the vehicle states from the number plate. The system was organized in this work in an echelon network of feature detectors that conducted consecutive processing of visual data pertaining to the dominant visual processing experience of the visual cortex, which influenced the computational model of the CNN. The findings of this research were observed with fewer training samples and turned out to be as 90 percent higher precision rate [3].

Andrew S. Agbemenu in 2018 proposed an ANPR method based on the characteristics and variations of the plates therein. The author has proposed in this work an algorithm that is enhanced to perform with Ghanaian license plate for conveyance. The designed model used two candidate detection algorithms as the detection of edges and the algorithms matching the template. The device then implemented the character segmentation technique, particularly with square plates to prevent noise effects, arrangement of characters, and skewing. At the final point, character recognition was rendered with the use of Tesseract OCR engine. Feature detection was slightly low but had a good success rate, with an average speed of 0.185s detecting 454 plates with 90.8 percent accuracy. The optical character recognition provided an average of 0.031s for the procedure and successfully identified approximately 60 percent of the detected plates [4].

Rayson Laroca in 2018 proposed an ALPR system that discussed the robustness and effectiveness of a framework based on the state-of-the-art YOLO artifact detector. The CNN are qualified and adapted for each ALPR stage to be resilient under different conditions. In this work, the author developed a two-stage attempt explicitly for the segmentation and identification of characters, using simple data augmentation artifices such as inverted number plates and character returns. The findings for the UFPR-ALPR dataset were found to be difficult as both commercial systems were below recognition levels of 70 percent. Yet the result was higher with a recognition rate of 78.33 percent for the proposed system [5].

Prashengit Dhar in 2018 developed an automated LPR program to support ITS for the identification of Bangladeshi license plates. This work plate shows clearly white background with black fonts. Prewitt operators performed the detection of the number plate to segment the edges. Morphological dilation was performed to accentuate the points. Eventually, deep CNN was used to accomplish the reconnaissance job. In character classification, the protocol showed a strong precision rate of 99.6 percent [6].

Cheng-Hung Lin in 2019 proposed a three-stage license plate recognition system based on Mask-RCNN which was used for various shooting angles and numerous oblique images. The author used YOLOv2 for the associated conveyance in the preceding stage for vehicle detection. The next stage was the location of the license

plate where YOLOv2 was again performed to detect the number plate. During this phase, YOLOv2 separates the images of phase I captured vehicles into 19 x 19 grids. In the final step, the author used Mask R-CNN for character recognition. The results in this work depict that the proposed model could classify vehicle number plates including bevel angles above 0-60 degrees and further accomplished the map rating of around 91 percent [7].

Nazmus Saif in 2019 has proposed a system to detect and recognize the Bangla license plate from the vehicle picture by using the convolutional neural networks. In this work, main focus to choose convolution neural network in the designed system is preferred because of its configuration for the end-to-end pipeline. CNN clearly outperformed conventional image processing algorithms for their case, and compared generalized CNN models better in different scenarios. The detection research was done using YOLOv3 which consists of 53 convolutional model layers. The second stage after identification is image segmentation and recognition of the characters it is. During this step, the device whips out the number plate region and then moves it to the second YOLO model for segmentation and platform image recognition. As a result, the model was checked with 200 images and correctly recognized the license plate number for 199 images, i.e. 99.5 per cent accuracy rate [8].

3. Problem Statement

There is an escalating increase of contemporary local, urban and national road networks over the last decades. This has emerged the need for efficient monitoring and management of road traffic. The goal of this project is to create a model that will be able to recognize and determine the number plate from its image appropriately. Due to the varying characteristics of the license plate from country to country like numbering system, colors, language of characters, style (font) and sizes of license plate, further research is still needed. The major goal of the proposed system is understanding Convolutional Neural Network, and applying it to the number plate recognition system.

4. Deep Learning Models

The current standard for dealing with medical photos is called "deep learning." It is meant to help radiologists make more accurate diagnoses by giving a quantitative analysis of potentially worrying lesions and making the clinical workflow go more smoothly. The goal is to get both of these benefits. Deep learning has already shown that it can do better than humans at tasks like recognition and computer vision. The structure of the deep learning algorithm is much more complicated than the structure of the standard method (machine learning). Pre-processing, enhancement, and inference are the three main steps that make up DL architecture. Pre-processing is the first step in DL architecture. In the second stage, different methods are used to pull out the characteristics of the input. In the third step of the procedure, the process of classifying each input based on a number of different classifiers is looked at. that a deep learning model doesn't need much help from a person, can deal with complex data that can be hard for machine learning, and gives correct results in a relatively short amount of time [28].

CNN Architecture- CNN is an example of DL architecture. It is usually used to solve problems with how pictures are organized, and it is very good at managing DL technology. The CNN was built on top of the artificial neural network (ANN) that has been used for a long time. The main goal of the ANN is to apply repeating patterns to a wide range of subfields in the industry of image modelling. Hierarchical neural networks are better than traditional feed-forward neural networks because they use a hierarchical method, which reduces the number of structural pieces needed (the number of artificial neurons). CNN uses both a feed-forward method and a very good way to find things. The network is easy to put together, and the user doesn't have to do much to set up the training settings. As an example, think about how much progress has been made in detecting things thanks to CNNs. On the other hand, problems that are often caused by a model that is hard to understand or has a lot of weights are greatly reduced. CNN has been shown to have a basic structure with five layers: an input layer, a folding layer with a setting function, a pooling layer, a completely related layer, and a SoftMax layer [29].

VGG16-VGG16 is an independent complex model projected by K. Simonyan and A. Zisserman from the University of Oxford in their paper "Deep convolutional neural networks for large image recognition". The model realizes an exam score of 5 for 92.7% on ImageNet, which has a database of more than 14 million images in 1,000 categories.

ResNet50-ResNet is not like other network topologies such as AlexNet, OverFeat, or VGG. ResNet is an example of a "exotic architecture" that is based on a micro-architecture model, also referred to as a "in-network architecture."

InceptionV3 - First presented the "creative" horizontal. In his 2014 paper Going Deeper with Convolutions: The goal of start-up model is to function as a "multi-feature explorer" by calculating 1×1 , 3×3 , or 5×5 convolutions. In the same model of the network- these the filter results are applied according to the dimensions of the channel. To improve the capacity of deep neural networks before use, the most direct way is to increase the depth of the network. However, as the depth of the network width increases, there are too many internal dimensions, which results in more resource consumption. Therefore, in order to overcome these problems, Szegedy et al first introduced the Inception model into the GoogLeNet architecture. 35 And completed an impressive performance and record as the winner of the ImageNet ILSVRC Challenge. The first model consists of an upper water-supply layer and a corresponding plate. The sizes of the mixed layers were 1×1 , 3×3 , and 5×5 , which were combined. Between two matching 1×1 layers, the maximum reduction is used to reduce the dimensionality and a tandem filter is required to combine the different layers. In addition, by removing the 5×5 convolutions and introducing two 3×3 convolutions to modify the initial model change, it was widely used in later network configurations.

MobileNetv2-The Mobile Net model is built on a tangible deep convolution, a type of broken convolution that converts a regular convolution to a deep complexity and a one-to-one complexity termed an effective convolution point. Convolution filters each communication channel in Mobile Nets using a single filter. After that, the wise complexity performs a one-to-one convolution to combine the deep convolution advances. Traditional convolution not only filters use, but combines the use of a series of new products in one process. The single column divides it into two layers, one for filtering and one for mixing. This deletion has an effect on the calculation of the calculation and the size of the model.

DenseNet - One of the recent breakthroughs in neural networks for the recognition of visual objects is referred to as DenseNet. DenseNet is quite similar to ResNet, although there are a few key distinctions between the two. DenseNet concatenates (.) the output of the previous layer; whereas ResNet employs an additive approach (+) that combines the identity of the layer that came before it with the identity of the layer that will come after it. Through the utilization of composite function operation, an output from the layer below functions as an input for the layer above it. The convolution layer, the pooling layer, the batch normalization, and the non-linear activation layer are the constituent parts of this composite procedure. Because of these links, the network has an absolute maximum of $L(L+1)/2$ direct connections. The number of levels in the architecture is denoted by the letter L. The DenseNet has different variants, such as DenseNet-121, DenseNet-160, DenseNet-201, etc. The digits represent the number of layers that are present in the neural network.

GoogLeNet The convolutional neural network known as GoogLeNet consists of a total of 22 layers. You have the option of loading a network that has already been trained, and it can be trained using either the ImageNet [1] or Places365 [2] [3] data sets. The network that was trained on ImageNet is able to classify images into one thousand different object categories, including things like a keyboard, mouse, pencil, and a wide variety of animals. The Inception architecture is the foundation for the GoogLeNet variety of convolutional neural network that Google developed. It does this through the utilization of Inception modules, which provide the network the ability to select from a number of different convolutional filter sizes in each block. These modules are layered one atop the other in an Inception network, which also includes some max-pooling layers with stride 2 now and then to reduce the resolution of the grid by one half.

5. Datasets And Methodology

The information came from local sources and the website : <https://www.kaggle.com/code/sankalpsinghbais/automatic-number-plate-recognition> the aim of this research is to design and analyze the License Plate Identification program mediated through Digital Images or Automatic Number Plate Recognition (ANPR), especially by using desktop peripheral. In doing so, license plates attached, especially, on cars will be the test subject of this research. Detailed elaboration on the designing process of the system that is going to be analyzed will be provided, and it includes the flowchart and the working procedure of the system. The parameters that are going to be measured within the designing process of the Automatic Number Plate Recognition program will also be provided. There are three particular steps that include Capture, Plate

Detection, and Plate Recognition 4,6 which are prioritized in the designing process of the ANPR program. Capture is the first step of the whole process, and it will highly influence the ANPR system. The quality of the image will influence the Plate Detection process. It will process the image until the excesses are removed, and it will leave the image of the actual license plate. Thus it will make thing easier for the Plate Recognition process.

6. Training And Evaluation Phases

The first step is called "Basic Training," and it involves training all of the layers of the complex from the bottom up. We can either randomly set up all of the layers or start from scratch and train them. This kind of training takes a long time to get right, but in the end, the results are very accurate. The letter B will stand for this stage of getting ready. Fine Tuning In this training process, we don't change the overall weight of the given picture in the convolutional layer. To put it simply, we just mix up the weights of layers that are close to each other. Then, training brings all of the layers together at the junction. It is important to remember that the dense layer is taught based on the weights that were chosen at random, while the convolutional layer is taught based on the net weights of the picture that is being trained on. This method is called "FT," which stands for its initials. With this method, the CNN's convolutional layer is never trained at any point during the learning process. Instead, we will keep the weights that were taught when the picture network was created. Here, you can only train with thick layers of weights that are set up in different ways.

During the training process, we make use of the algorithms known as Stochastic Gradient Descent with Momentum (SGDM), Adaptive Moment Estimation (ADAM), and Root Mean Square Propagation (Rmsprop). Research aims necessitate the observation of performances. During this stage of the procedure, we will begin training all of the layers of the complex from the ground up. We choose to either randomly initialize all of the layers or train them from scratch. This type of training takes a very long time to converge, but it ultimately results in fairly accurate results. Approaches based on SGDM and its many offshoots have proven to be the most successful in helping students overcome severe learning obstacles (including deep learning). The accumulation of momentum is a process that can either contribute to the quickening of the SGD through the control route or the dampening of oscillations. It was decided that the working time would be 0.9. As a direct consequence of this, momentum is built up at a faster rate, and oscillations end up being more tightly packed.

Table 1: Deep Learning Models and Number of Parameters

Deep learning models	Pixels	parameter
VGG16	224, 224, 3	138 million
GoogLeNet	224, 224	62.3 million
Inceptionv3	224, 224	24 Million.
MobileNet	1, 224×224	13 million
ResNet	224, 224, 3	23 million
DesNet	224, 224, 3	36928M

7. System Model

The overall ANPR system can be subdivided into the software model and hardware model. The section will discuss both models in detail. A. Software Model main and most important portion of this system is the software model. The software model uses series of image processing techniques which are implemented in MATLAB 19b. The ANPR algorithm is broadly divided into three parts:

- Capture image
- Extract the plate from the image
- Recognize the numbers from the extracted.

The images are captured in RGB format so they can be further processed for the number plate extraction. The second step of the ANPR algorithm is the extraction of the number plate in an image. A background colour search algorithm is used to extract the likelihood ROI in an image. As the official number plate has colour background with alphanumeric characters written in black, it is easy to detect the plate area by searching for specific colour pixels. The image is search for the background color pixels or some which are closer to background colour in value. If pixel value is of background color the pixel is set to 1, otherwise the pixel value is set to 0. The image obtained after the search algorithm is in black and white format. After identify the ROI, image is then filtered using two different filtering techniques. The first technique involves removing of all white patches that are connected to any border and set their pixel value to 0. The second filtering technique use pixel count method to remove the small regions in an image other than the plate region. The number of consecutive white pixels is inspected and regions that contain number of white pixels less than the predefined threshold are set to 0. At this stage the image contains only the vehicle number plate.

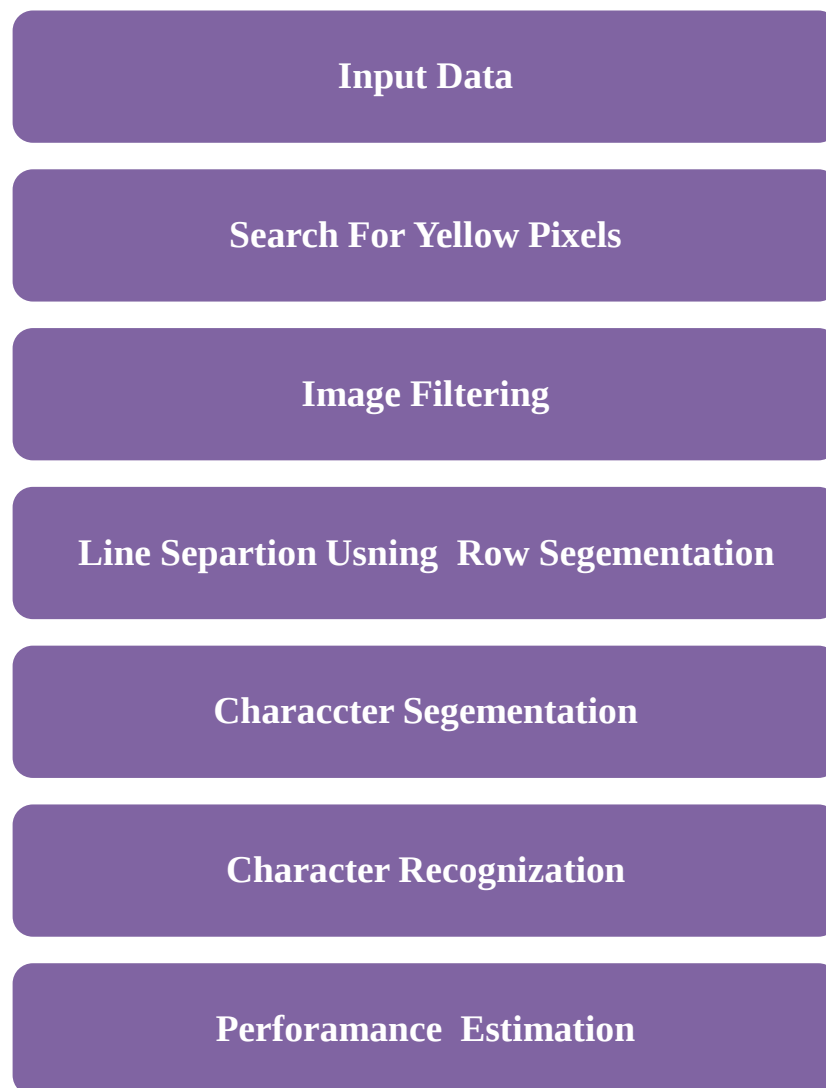


Fig 2: Flow diagram

The image is then cropped that only contains the vehicle number plate. The third step of the developed ANRP algorithm uses the Optical Character Recognition (OCR) algorithm to recognize the vehicle number. The resultant cropped image obtained after the second step is inverted i.e. all white pixels are converted to black and black pixels to white. Now the text is in white and the plate background is black. Before applying the OCR the individual lines in the text are separated using line separation process. The line separation adds each pixel's value in a row. If the resultant sum of row is zero that means no text pixel is present in a row and if the resultant sum of the row is greater than zero that means the text is present in row. The first resultant sum greater than zero represents

the start of the line and after this, the first resultant sum equal to zero represents the end of the line. The start and end values of the line are used to crop the first line in the text. The same process continues to separate the second line in the text. Once the lines in an extracted vehicle number plate are separated, the line separation process is now applied column-wise so that individual characters can be separated. The separated individual characters are then stored in separate variables. The OCR is now used to compare each individual character against the complete alphanumeric database.

A-Vehicle image capture

The initial step is the Acquisition of an image i.e., getting an image using the digital camera associated with the PC. These Caught images are in RGB format so it can be further processed for the Number Plate Extraction. The database system contains the personal information of the vehicle proprietor and a few plate vehicle images, abbreviations, and acronyms.

B-Pre-processing

The captured RGB image is appeared in fig-4. The captured image is influenced by many elements like: Optical system distortion, system commotion, lack of presentation, or over-the-top relative motion of camera or vehicle, and so forth result is the degradation of a captured vehicle image and the unfriendly influence to the further image processing. Therefore before the main image processing, pre-processing of the captured image is taken out which includes converting RGB to gray, clamor evacuation, and border enhancement for brightness as shown in fig 5.



Fig 3: Captured image by digital camera



Fig 4: RGB to gray converted image



Fig 5: Vehicle Number Plate extraction

C-Number plate Extraction

The Basic step in extraction of number plate is to identify the plate size and large number of plates are rectangular in shape. Since, number plates are rectangular we can use shape as a key point to extract the number plate by eliminating the vehicle as shown in fig 6.

D-Character Segmentation

Character segmentation is an operation that seeks to decompose an image of a sequence of characters into sub images of individual symbols. It is one of the decision processes in a system for optical character recognition (OCR). Its decision, that a pattern isolated from the image is that of a character (or some other identifiable unit), can be right or wrong. The image is looked for any pixels that satisfy the necessities. At whatever point such a pixel is experienced, its neighbors are checked, and if any of the neighbors likewise parallel the criteria, both the pixels are measured as have a place with the same region. We get individual character and number image by using, vertical and horizontal scanning technique. Fig 7 demonstrates the plate segmentation case.

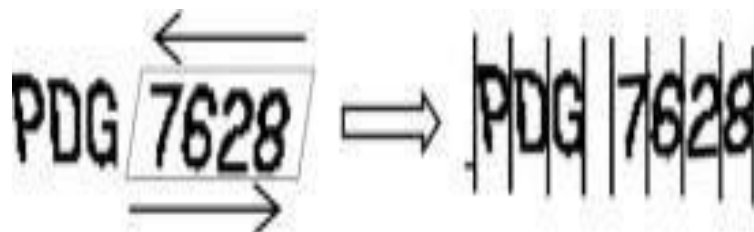


Fig 6: Example of Plate Segmentation.

E-Character Recognition

This is the most essential and basic phase of the ANPR system. It displays the techniques that were used to order and then perceive the individual characters. The classification is based on the extracted features. Character recognition is the optical character recognition (OCR) is used to look at every individual character against the complete alphanumeric database. The OCR really uses relationship strategy to match individual character and finally, the number is recognized and stored in string format in a variable. The character is then contrasted and the database for the vehicle authorization. The resultant signs are offered according to the consequence of comparison. Templates will exist for every one of the characters i.e. A-Z and 0-9 as appeared in figure fig 8.



Fig 7: Character Recognition

8. Training Methods

The ADAM algorithm is an example of an algorithm that helps to improve something. It could take the place of the stochastic gradient ancestor when the network value is being updated. This method is used to figure out the right learning rate for each parameter. Adam also remembers how much damage gradients have usually done in the past, which is the same thing as momentum. The Adam algorithm is used a lot in the field of deep learning because it can produce high-quality results in a short amount of time. A learning rate parameter is given to RMSProp, which is then changed based on the average of the most recent light weight magnitude. This shows that the method can be used to solve problems that have nothing to do with the Internet. RMSprop divides by dividing the learning rate by the average amount of damage that comes from a square gradient. [31]

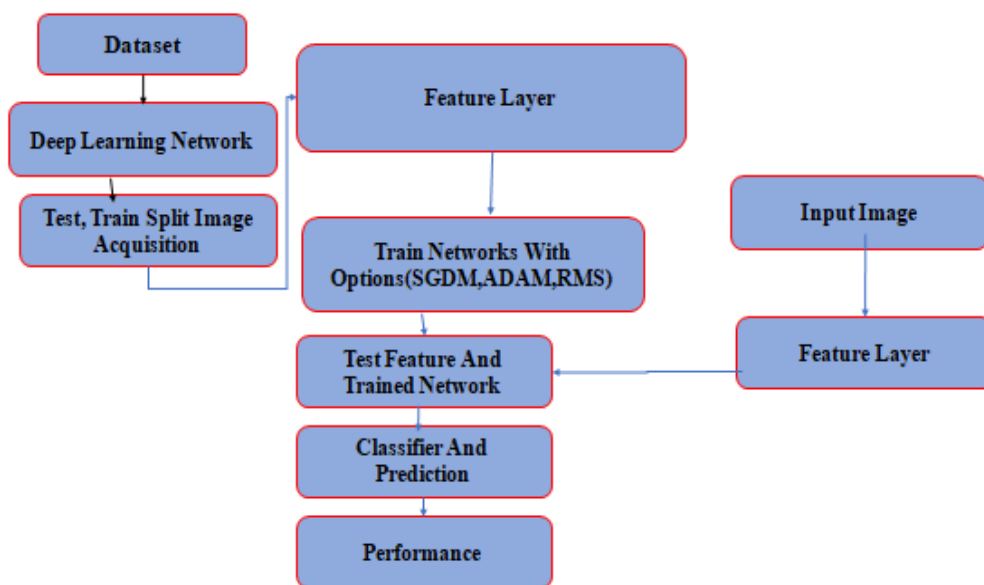


Fig 8: training and testing flow of proposed system

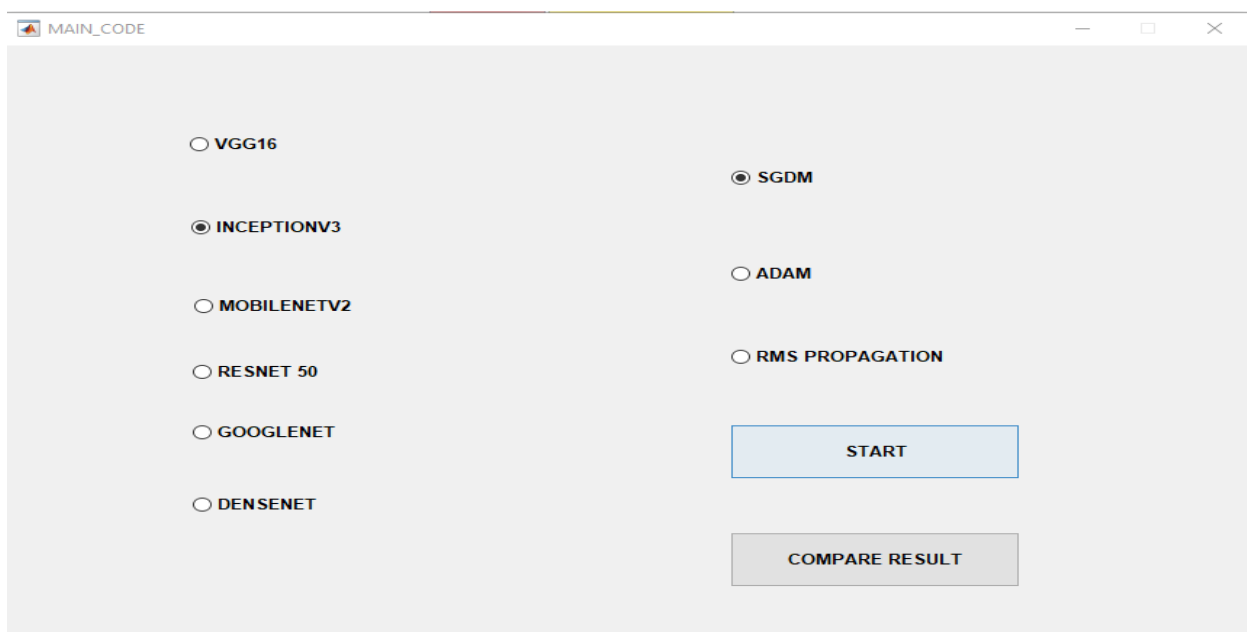


Fig 9: proposed pre-trained deep learning models

9. PROPOSED WORK

The number of vehicles on the road has exponentially increased which are the major reasons for traffic congestion and violation. In order to reduce the violations and also to automate traffic management, ANPR (Automatic Number Plate Recognition) has been developed. There are various ANPR techniques carried out in India and their efficiency can be increased. The proposed system focuses to optimize and improve the efficiency of ANPR. Initially the system was trained with pre-trained deep learning algorithm. For segmenting the characters, the region of characters should be enhanced and for this histogram equalization (also known as gray level scaling) is used. By application of enhancement of the region of characters, there is also enhancement in noise [2]. For segmenting the correct characters, only character pixels are needed and can be enhanced, however, the pixels of the background remained to be depleted

Algorithm Steps:

- Step 1: Input Image of number plate
- Step 2: It is loaded (Image) classified as being ONE, TWO or THREE
- Step 3: Preprocess() on number plate image
- Step 4: It is determined the a priori probability for each class of digit and character
- Step 5: Remove() all background data.
- Let (i,j) It is calculated the words and number.
- Step 6: Clustering () of words character and digit details.
- Step 7: Recognition() all words.
- Step 8: Stop



Figure 10



Figure 11

10. Experiments And Results

We used DL algorithms on the dataset of ANPR that we got to figure out how bad the symptoms were. We used the programming language Matlab/Python to make the deep learning algorithms we chose, which were Inceptionv3, Vgg16, Resnet50, DenseNet, GoogLeNet, and Mobilenetv2, so that we could put the dataset into groups. As you can see in Table 2, we used precision, recall, f1 score, and accuracy as metrics to show how well and reliably each approach worked. Where accuracy is defined as the ratio of true positives to the total number of true positives plus false positives. The sensitivity is found by taking the average of the recall and accuracy scores. The recall score is the number of real positives compared to the total number of positives and negatives, including false positives and false negatives. The accuracy score is the number of correct answers. When putting patients' cases into groups based on the long-term conditions they had, these algorithms gave us enough information. In Table 2, the results of each method are shown, along with their ratings for accuracy, precision, recall, sensitivity, specificity dice coefficient and Jaccard coefficient score.

Table 2: Comparison of Pre-trained Models

Pre-Trained Model	Learning Techniques	Training Techniques	Validation Accuracy(%)
Inceptionv3	Baseline Learning	RMS Propagation	91.33%
		SGDM	89.11%
		ADAM	81.15%
VGG 16	Baseline Learning	RMS Propagation	90.29%
		SGDM	72.32%
		ADAM	91.82%
ResNet50	Baseline Learning	RMS Propagation	94.85%
		SGDM	93.77%
		ADAM	94.78%
DenseNet	Baseline Learning	RMS Propagation	90.24%
		SGDM	92.36%
		ADAM	93.37%

GoogleNet	Baseline Learning	RMS Propagation	90.75%
		SGDM	92.17%
		ADAM	92.52%
MobileNetv2	Baseline Learning	RMS Propagation	91.22%
		SGDM	91.27%
		ADAM	92.44%

Table 2 shows the results, which show that the baseline learning technique and the Inceptionv3 model with SDGM training method have an accuracy of 95.25%, ADAM has an accuracy of 80.21%, and RMS Propagation has an accuracy of 90.22%. The SDGM training technique gave a 93.48% accuracy for the vgg16 model, the RMS Propagation method gave 94.91% accuracy for the ADAM model, and the RMS Propagation method gave a 93.48% accuracy. Using the SDGM training method, the resnet50 model has reached 89.80% accuracy, the ADAM model has reached 95.3% accuracy, and the RMS Propagation model has reached 93.11% accuracy. Using the SDGM training method, the DenseNet model's accuracy has reached 96.32%, the ADAM model's accuracy has reached 94.56%, and the RMS Propagation model's accuracy has reached 92.12%. Using the SDGM training method, the GoogleNet model's accuracy has reached 90.25 percent, the ADAM model's accuracy has reached 96.32%, and the RMS Propagation model's accuracy has reached 89.15 percent. The accuracy of the MobileNetv2 model trained with the SDGM training method has reached 95.56 percent, the accuracy of the ADAM model has reached 90.25 percent, and the accuracy of the RMS Propagation model has reached 98.69 percent.

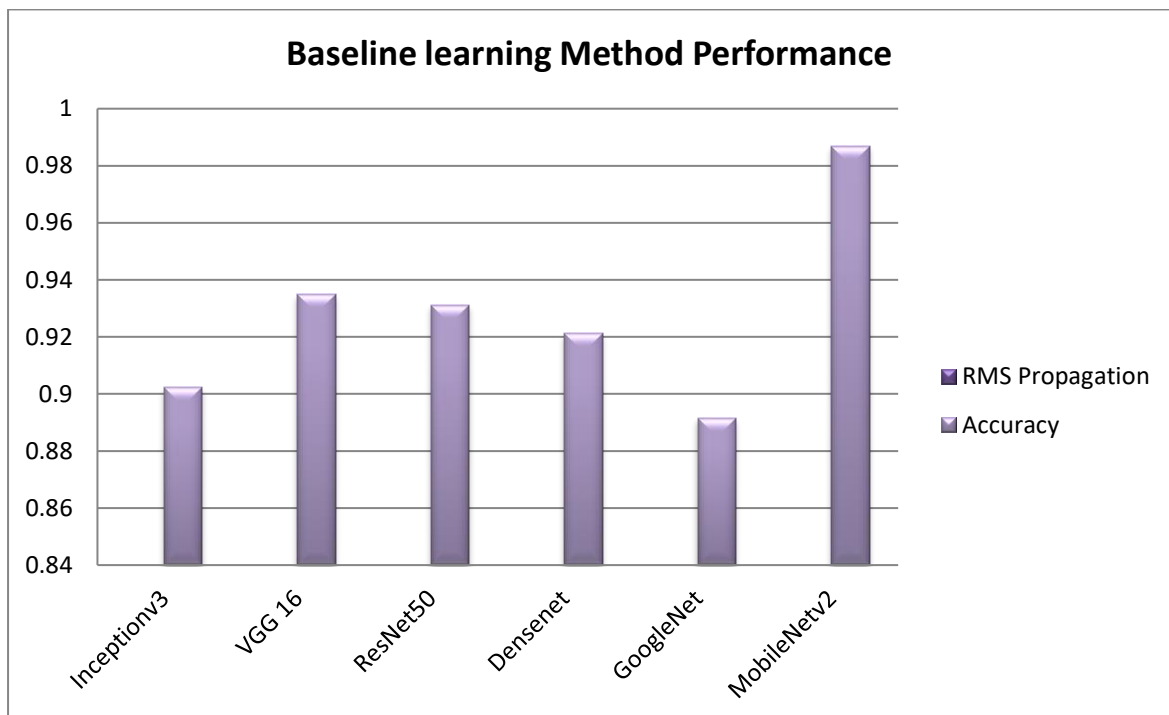


Fig 12: Baseline Learning and RMS Propagation Training Method performance

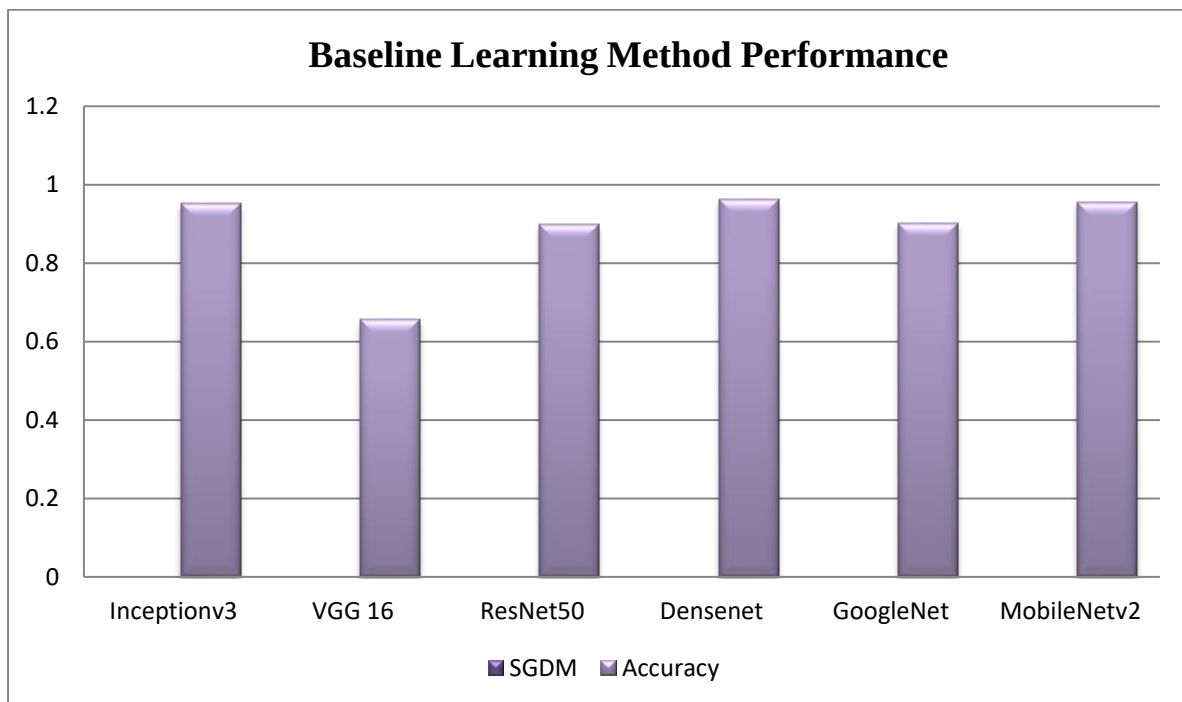


Fig 13: Baseline Learning and SGDM Training Method performance

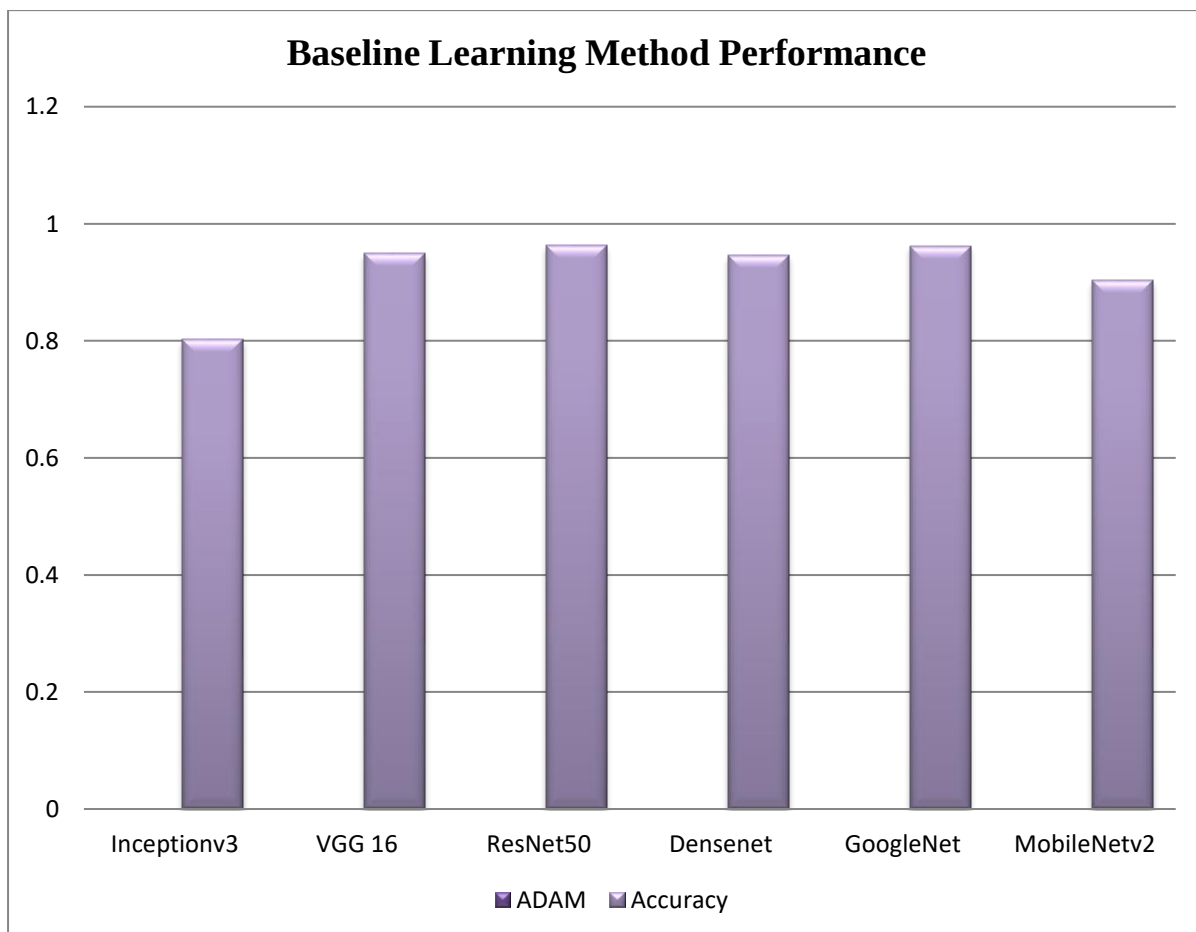


Fig 14: Baseline Learning and ADAM Training Method performance

Table 3: classification and Segmentation Performance for SGDM Training Techniques

Pre-Trained Model	Sensitivity	Specificity	Precision	Recall	Jaccard Coefficient	Dice Coefficient
InceptionV3	90.33%	91.22%	90.88%	91.92%	91.02%	88.23%
VGG16	92.17%	90.88%	91.24%	89.26%	90.28%	92.29%
ResNet50	94.56%	94.66%	94.62%	93.92%	95.87%	94.77%
DenseNet	91.77%	88.64%	89.15%	92.65%	85.89%	91.56%
GoogleNet	91.35%	88.92%	91.24%	90.34%	90.75%	91.47%
MobileNetv2	87.98%	90.99%	92.98%	89.36%	89.66%	91.34%

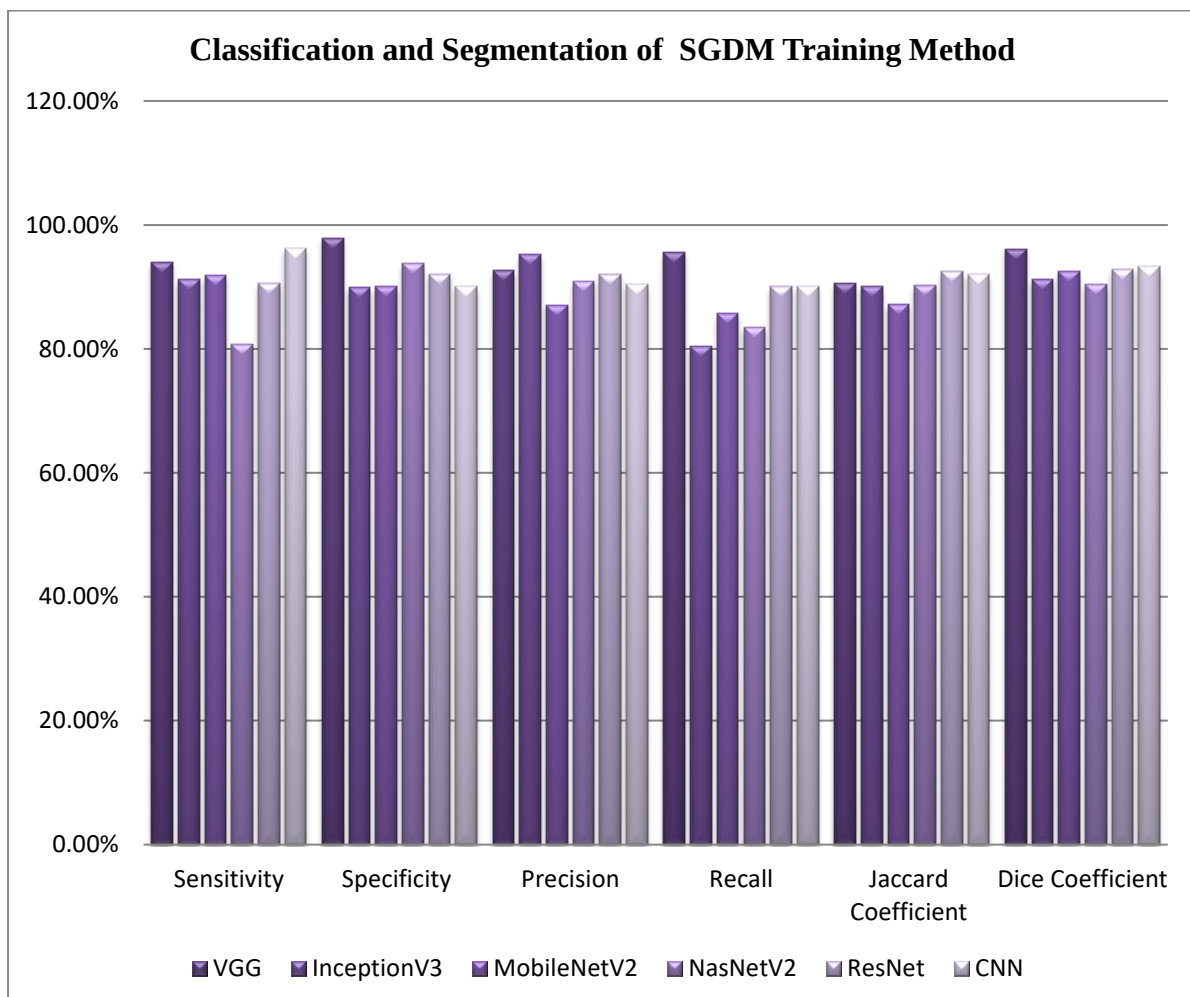


Fig 15: showing the Classification and Segmentation Performance for SGDM Training Method

Table 4: showing the classification and segmentation performance for ADAM Training Method

Pre-Trained Model	Sensitivity	Specificity	Precision	Recall	Jaccard Coefficient	Dice Coefficient
InceptionV3	91.32%	92.55%	91.34%	90.10%	91.98%	92.57%
VGG16	91.70%	87.93%	89.33%	89.65%	91.87%	92.92%
ResNet50	95.95%	95.87%	94.55%	94.88%	95.77%	94.89%
DenseNet	82.88%	88.55%	89.24%	87.87%	91.21%	89.28%
GoogleNet	90.26%	91.11%	91.26%	92.38%	92.97%	89.28%
MobileNetv2	92.35%	92.55%	93.89%	91.25%	92.65%	91.98%

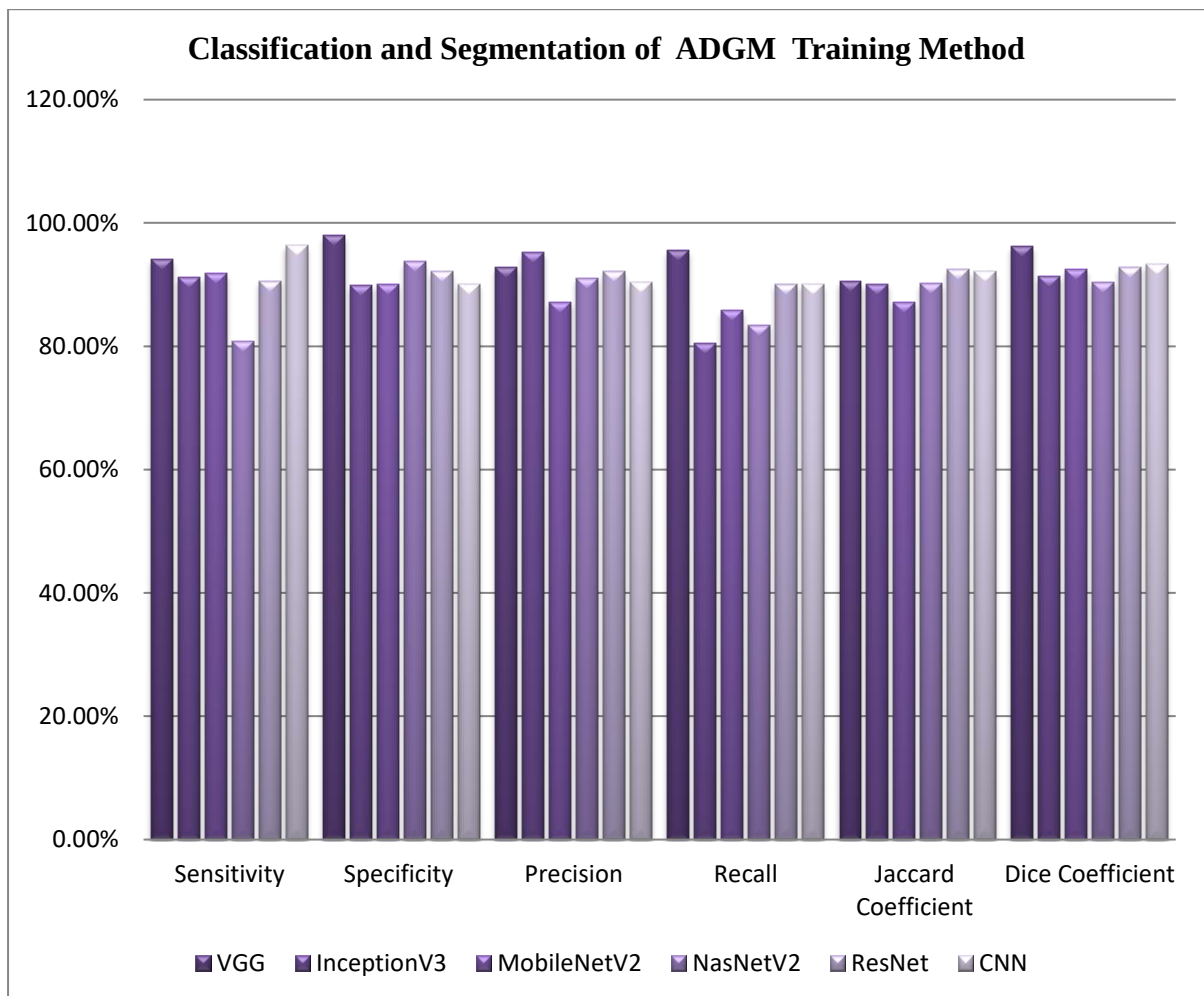


Fig 16: showing the Classification and Segmentation Performance for ADAM Training Method

Table 5: showing the Classification and Segmentation Performance for RMS Propagation Training Method

Pre-Trained Model	Sensitivity	Specificity	Precision	Recall	Jaccard Coefficient	Dice Coefficient
InceptionV3	94.25%	95.90%	93.71%	98.31%	95.54%	95.20%
VGG16	91.15%	80.89%	95.23%	86.40%	90.14%	91.33%
ResNet50	95.85%	96.12%	87.05%	95.23%	94.12%	96.50%
DenseNet	85.78%	94.78%	90.98%	84.45%	90.33%	90.32%
GoogleNet	94.60%	95.12%	92.12%	94.13%	94.43%	92.45%
MobileNetv2	91.85%	94.12%	87.305%	84.23%	84.12%	92.50%

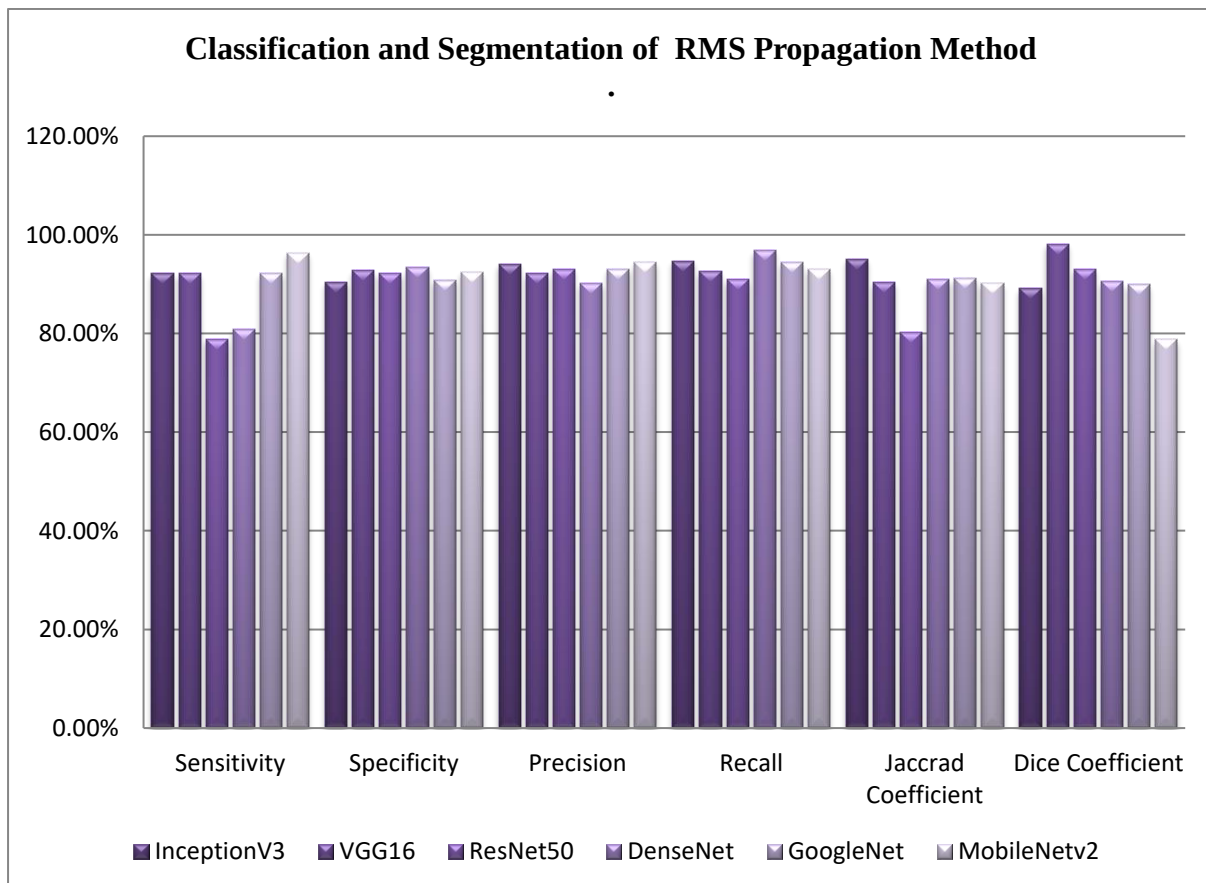


Fig 12: showing the Classification and Segmentation Performance for RMS Propagation

Training Method From Table 3,4,5 Showing The RMS Propagation SGDM, ADAM Training Method realize the best performance in terms of accuracy, precision, specificity, sensitivity, Recall, Jaccard Coefficient, Dice Coefficient and with values Showing In The Table 3,4,5

Table 8: proposed Pre-trained models comparison with existing techniques

Studies	DL Models	Accuracy
Kashyap, A et al.[29].38	Edge statistics and morphology techniques	75%
Kraisin, S et. al.140	Used provided Images	90%
Vaishnav, A et. al.141	Morphology Techniques	.95%
Sferle, R.M. et al.[32]. 36	Histogram Analysis using HOG	89%
Haider, S.A. et. al.[27] 23	Histogram Analysis using HOG	90%
Proposed Pre-Trained Models	Inceptionv3	94.23%
	Vgg16	94.30%
	Resnet50	96.12%
	DenseNet	94.25%
	GoogLeNet	92.12%
	Mobilenetv2	98.69%

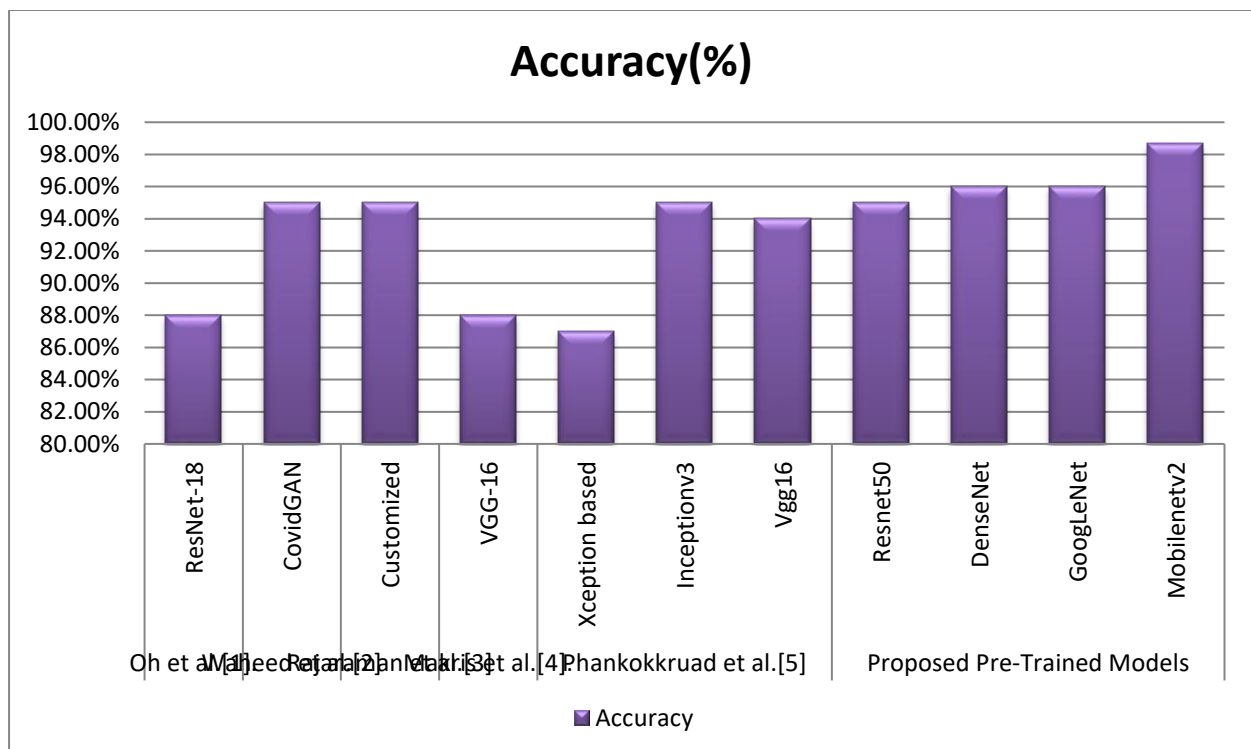


Fig 18: proposed Pre-trained models comparison with existing techniques

11. Conclusion

The approach of vehicle number plate recognition needs a very high degree of accuracy when we are performing our task on a very busy road or parking area which may not be possible manually. As a human beings influenced to get fatigued due to the repetitious nature of the job and they cannot keep a record of the vehicles when there are large numbers of vehicles are passing in a very short time. To avoid this problem, many attempts have been simulated by the researchers worldwide the globe for the last many years. A similar attempt has been made in this work to develop a specific and automatic number plate recognition system. The accuracy of the

proposed system, can be useful for vehicle identification as the average recognition rate is about 95% on combined data. It may be achieved that the project has been by and far fruitful. For future studies online cautioning of vehicles can be done in case of stolen vehicles.

It consists of the outcomes of the result analyzed based on the various parameters such as recognition of individual characters and a digit success ratio of recognition for the success of identifying selected set of a character from group of characters and digits. From the above results, we can conclude that number plate recognition will perform better as the quality of the camera used for scanning the plate will be excellent. Using low-quality camera will degrade the performance and may misclassify the characters.

With the increase in the number of vehicles, vehicle tracking has become an important research area for efficient traffic control, surveillance, and finding stolen cars. For this purpose, efficient real-time license plate detection and recognition are of great importance. Due to the variation in the background and font color, font style, size of the license plate, and non-standard characters, license plate recognition is a great challenge in developing countries. To overcome such issues, this study applies a deep-learning strategy to improve license plate recognition efficiency. The collected images have been captured under various lighting/contrast conditions, distance from the camera, and varying angles of rotation, and validated to produce a high recognition rate. The approach can be effectively used by law enforcement agencies and private organizations to improve homeland security. Future work may include training and validation of the existing algorithm using the hybrid classifier method and improvement of the robustness of the license plate recognition system in varying weather conditions.

ANPR can be further exploited for vehicle owner identification, vehicle model identification traffic control, vehicle speed control, and vehicle location tracking. It can be further extended as multilingual ANPR to identify the language of characters automatically based on the training data. It can provide various benefits like traffic safety enforcement, security- in case of suspicious activity by vehicle, easy to use, immediate information availability- as compare to searching vehicle owner registration details manually and cost-effective for any country. For low-resolution images some improvement algorithms like super-resolution of images should be focused. Most of the ANPR focus on processing one vehicle number plate but in real-time there can be more than one vehicle number plates while the images are being captured.

Reference

- [1] Kashyap, A.; Suresh, B.; Patil, A.; Sharma, S.; Jaiswal, A. Automatic number plate recognition. In Proceedings of the Communication Control and Networking (ICACCCN), Greater Noida, India, 12–13 October 2018; pp. 838–843.
- [2] Kraisin, S.; Kaothanthong, N. Accuracy Improvement of A Province Name Recognition on Thai License Plate. In Proceedings of the 2018 International Joint Symposium on Artificial Intelligence and Natural Language Processing (ISAI-NLP), Pattaya, Thailand, 15–17 November 2018; pp. 1–6.
- [3] Vaishnav, A.; Mandot, M. Integrated automatic number plate recognition for recognizing multi language fonts. In Proceedings of the 2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), Noida, India, 29–31 August 2018; pp. 551–556.
- [4] Sferle, R.M.; Moisi, E.V. Automatic Number Plate Recognition for a Smart Service Auto. In Proceedings of the 2019 15th International Conference on Engineering of Modern Electric Systems (EMES), Oradea, Romania, 13–14 June 2019; pp. 57–60.
- [5] Haider, S.A.; Khurshid, K. An implementable system for detection and recognition of license plates in Pakistan. In Proceedings of the 2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT), Karachi, Pakistan, 5–7 April 2017; pp. 1–5.
- [6] Rasheed, S.; Naeem, A., & Ishaq, O. (2012, October). Automated number plate recognition using hough lines and template matching. In Proceedings of the World Congress on Engineering and Computer Science (Vol. 1, pp. 24-26).
- [7] Sasi, A., Sharma, S., & Cheeran, A. N. (2017, March). Automatic car number plate recognition. In 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS) (pp. 1-6). IEEE.
- [8] Subhadhira, S., Juithonglang, U., Sakulkoo, P., & Horata, P. (2014, March). License plate recognition

- application using extreme learning machines. In 2014 Third ICT International Student Project Conference (ICT-ISPC) (pp. 103-106). IEEE.
- [9] Shima, Y. (2016, December). Extraction of number plate images based on image category classification using deep learning. In 2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS) (pp. 19-26). IEEE. Volume 6, Issue 3, May-June-2020 | <http://ijsrcseit.com> Shally Gupta et al Int J Sci Res CSE & IT, May-June-2020; 6 (3) : 955-966 966
- [10] Bhardwaj, E. D., & Gujral, E. S. (2014, June). Automated Number Plate Recognition System Using Machine learning algorithms (Kstar). In International Journal of Enhanced Research in Management and Computer Applications, ISSN:2319-7471 Vol.3 Issue 6, June2014,pp:(42-47), Impact Factor:1.147, Available online at : www.erpublications.com .
- [11] Liu, W. C., & Lin, C. H. (2017, May). A hierarchical license plate recognition system using supervised K-means and Support Vector Machine. In 2017 International Conference on Applied System Innovation (ICASI) (pp. 1622- 1625). IEEE.
- [12] Li, Y., Niu, D., Chen, X., Li, T., Li, Q., & Xue, Y. (2019, July). Vehicle License Plate Recognition Combing MSER and Support Vector Machine in A Complex Environment. In 2019 Chinese Control Conference (CCC) (pp. 7045-7050). IEEE.
- [13] Yang, G. (2011, March). License plate character recognition based on wavelet kernel LS-SVM. In 2011 3rd International Conference on Computer Research and Development (Vol. 2, pp. 222-226). IEEE.
- [14] Bhardwaj, D., & Kaur, H. (2014, October). Comparison of ML algorithms for identification of Automated Number Plate Recognition. In Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization (pp. 1-6). IEEE. [10] Satsangi, M., Yadav, M., & Sudhish, P. S. (2018, October). License Plate Recognition: A Comparative Study on Thresholding, OCR and Machine Learning Approaches. In 2018 International Conference on Bioinformatics and Systems Biology (BSB) (pp. 1-6). IEEE.
- [15] Ni, K., Fu, M., Huang, Z., & Sun, S. (2018, August). A Proposed License Plate Classification Method Based on Convolutional Neural Network. In 2018 International Conference on Network Infrastructure and Digital Content (IC-NIDC) (pp. 344-347). IEEE.
- [16] Selmi, Z., Halima, M. B., & Alimi, A. M. (2017, November). Deep learning system for automatic license plate detection and recognition. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR) (Vol. 1, pp. 1132-1138). IEEE.
- [17] Kim, K.K., Kim, K.I., Kim, J.B. and Kim, H.J., 2000, December. Learning-based approach for license plate recognition. In Neural Networks for Signal Processing X. Proceedings of the 2000 IEEE Signal Processing Society Workshop (Cat. No. 00TH8501) (Vol. 2, pp. 614-623). IEEE.
- [18] Han, C.C., Hsieh, C.T., Chen, Y.N., Ho, G.F., Fan, K.C. and Tsai, C.L., 2007, October. License plate detection and recognition using a dual-camera module in a large space. In 2007 41st Annual IEEE International Carnahan Conference on Security Technology (pp. 307-312). IEEE. 3. Mondal,
- [19] M., Mondal, P., Saha, N. and Chattopadhyay, P., 2017, December. Automatic number plate recognition using CNN based self-synthesized feature learning. In 2017 IEEE Calcutta Conference (CALCON) (pp. 378-381). IEEE.
- [20] Agbemenu, A.S., Yankey, J. and Addo, E.O., 2018. An automatic number plate recognition system using opencv and tesseract ocr engine. International Journal of Computer Applications, 180, pp.1-5.
- [21] Laroca, R., Severo, E., Zanlorensi, L.A., Oliveira, L.S., Gonçalves, G.R., Schwartz, W.R. and Menotti, D., 2018, July. A robust real-time automatic license plate recognition based on the YOLO detector. In 2018 International Joint Conference on Neural Networks (IJCNN) (pp. 1-10). IEEE.
- [22] . Dhar, P., Guha, S., Biswas, T. and Abedin, M.Z., 2018, February. A system design for license plate recognition by using edge detection and convolution neural network. In 2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2) (pp. 1-4). IEEE.
- [23] . Lin, C.H. and Li, Y., 2019, August. A License Plate Recognition System for Severe Tilt Angles Using Mask RCNN. In 2019 International Conference on Advanced Mechatronic Systems (ICAMEchS) (pp. 229-234). IEEE.

- [24] Saif, N., Ahmmed, N., Pasha, S., Shahrin, M.S.K., Hasan, M.M., Islam, S. and Jameel, A.S.M.M., 2019, October. Automatic License Plate Recognition System for Bangla License Plates using Convolutional Neural Network. In TENCON 2019-2019 IEEE Region 10 Conference (TENCON) (pp. 925-930). IEEE