Dissertation on

An Empirical Study on Traffic Sign Recognition using Convolutional Neural Networks

Thesis submitted towards partial fulfilment of the requirements for the degree of

Master in Multimedia Development

Submitted by

Sreejita Chakraborty

EXAMINATION ROLL NO.: M4MMD23005

UNIVERSITY REGISTRATION NO.: 160391 of 2021-2022

Under the guidance of

Mr. Joydeep Mukherjee

School of Education Technology

Jadavpur University

Course affiliated to

Faculty of Engineering and Technology

Jadavpur University

Kolkata-700032

India

2023

Master in Multimedia Development
Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata, India

CERTIFICATE OF RECOMMENDATION

This is to certify that the thesis entitled **"**AN EMPIRICAL STUDY ON TRAFFIC SIGN RECOGNITION USING CONVOLUTIONAL NEURAL NETWORKS**"** is a bonafide work carried out by SREEJITA CHAKRABORTY under our supervision and guidance for partial fulfilment of the requirements for the degree of Master in Multimedia Development in the School of Education Technology, during the academic session 2022-2023.

SUPERVISOR
School of Education Technology
Jadavpur University,
Kolkata-700032

DIRECTOR
School of Education Technology
Jadavpur University,
Kolkata-700032

DEAN - FISLM
Jadavpur University,
Kolkata-700032

Master in Multimedia Development
Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata, India

CERTIFICATE OF APPROVAL **

This foregoing thesis is hereby approved as a credible study of an engineering subject carried out and presented in a manner satisfactory to warranty its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not endorse or approve any statement made or opinion expressed or conclusion drawn therein but approve the thesis only for purpose for which it has been submitted.

Committee of final examination for evaluation of Thesis ----------------------------------

----

** Only in case the thesis is approved.

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis contains literature survey and original research work by the undersigned candidate, as part of her in **Master Multimedia Development** studies.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by this rule and conduct, I have fully cited and referenced all materials and results that are not original to this work.

NAME : ***SREEJITA CHAKRABORTY***

EXAMINATION ROLL NUMBER: ***M4MMD23005***

THESIS TITLE : ***An Empirical Study on Traffic Sign Recognition Using Convolutional Neural Networks.***

SIGNATURE: DATE:

Acknowledgement

I feel extremely glad in presenting this thesis at School of Education Technology, Jadavpur University, Kolkata, in the partial fulfilment of the requirements for the degree of Master in Multimedia Development.

I would like to express my sincere gratitude to my guide, Mr. Joydeep Mukherjee, for his vigilant supervision and constant encouragement. He spent her precious time in reviewing the research work and provided many insightful comments and constructive criticism. Without his enthusiasm, advice and unique support this research work would never have become a reality.

I am also grateful to Prof. (Dr.) Matangini Chattopadhyay, Director of School of Education Technology, for her support, encouragement and timely advices. I am really indebted to Dr. Saswati Mukherjee for her continuous support during the entire course of the research. Their advice and support were highly inspirational and motivating.

I would also like to take this opportunity to thank all of my classmates of Master in Multimedia Development course who motivated me to complete my research work successfully. I do wish to thank all of our departmental support staff and all of those who were associated with this research contributed in some form or the others.

Lastly but not least, I would like to acknowledge and thank my parents for the strong, unconditional support, inspiration and encouragement that they have provided me, without which I would be unable to complete this research.

With Regards,

DATE:

SREEJITA CHAKRABORTY
Master in Multimedia Development
School of Education Technology
Jadavpur University, Kolkata-700032

Dedicated to,

My Parents

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ABBREVIATION

ITS	Intelligent Transport Systems
TSD	Traffic Sign Detection
TSR	Traffic Sign Recognition
CNN	Convolutional Neural Network
R-CNN	Region-Based Convolutional Neural Network
LIDAR	Light Detection and Ranging
RGB	Red, Green and Blue
ReLU	Rectified Linear Unit
SSD	Single Shot MultiBox Detector
YOLOv2	You Only Look Once version 2

LIST OF FIGURES

Chapter 1
Executive Summary

This thesis presents a system for recognizing and classifying road and traffic signs. The purpose of this system is to develop an inventory of these signs, which can assist highway engineers in updating and maintaining them. Unveiling traffic signs (TSD), the main technology of TSR, is a significant issue due to various styles, small size, challenging riding conditions and obstacles.

At first, images were collected from online sources using devices such as cameras, mobile phones and electronic devices from a moving vehicle. Then it was pre-processed by resizing and converting to grayscale. This thesis is based on CNNs. As CNN requires input images to be a certain size and in grayscale, they must be pre-processed before being fed into the CNN module.

In the CNN module, the present researcher uses three convolution layers with 32 filters, 64 filters and 128 filters, respectively. Also, it has three max pooling layers and a ReLU activation layer.

Dropout layer of 0.5 is present to prevent the occurrence of overfitting problem.

Here in this thesis, the present researcher has utilized a substantial amount of data for both training and testing purposes. The present researcher has used a total of 39,209 images for training and 10,000 images for testing purposes. The present researcher has successfully achieved a 99.52% accuracy rate, which is an improvement compared to the previous work on Traffic Sign Recognition using Convolutional Neural Network.

Chapter 2

Introduction

In order to protect themselves and their pilgrims' assured outing, drivers must insist on releasing effectively at the right moment and in the right place. In any event, because to differences in environmental circumstances or survey centers, indications can be difficult to detect until the last defining moment has passed. A day's boost in computing power has quickly passed on to applications. However, the rise in fender benders in tandem with the growing volume of traffic has become a major concern for society [1].

The risk of street accidents is particularly high in areas with notable street features, such as the turn into a single-course route, steep bends and associations. Adding "STOP", "NO LEFT TURN", "NO ENTRY" and other signs to inform drivers of the road's conditions and other traffic information is one potential countermeasure. However, there is still a chance that the driver, who is relying on his or her perspective, will fail to see the sign while driving and a serious accident is possible if the driver fails to see a sign that reads "Don't ENTER", "STOP", or similar [2].

By applying an adjusted sign board assertion framework to provide traffic data to the driver and recall information about the street in front of the vehicle, accidents may be avoided. Additionally, signs have distinct shapes like circles, triangles, squares and octagons. Drivers can travel more safely thanks to these structures. The driver of the car is alerted while driving, saying, "Go moderate; ahead is a speed breaker" [3].

There are many revelation systems designed specifically for traffic light and sign board unequivocal confirmation. There is no framework that connects the disclosure layout of traffic signs and sending the alert message [4].

This thesis addresses road and traffic signage that use a visual/symbolic language regarding the roads ahead that drivers may interpret. In this thesis, the phrases are used interchangeably and they may appear in combination elsewhere as "road traffic signs." They give the motorist information that makes driving safer and more convenient. A sort of sign that is NOT considered in this thesis is the direction sign, which shows the impending instructions for travelling to named towns or on numbered routes using text rather than symbols.

Road and traffic signs must be appropriately erected in the appropriate locations and an inventory of them is preferably required to aid in adequate updating and maintenance. Automatic traffic sign detection and recognition can make a substantial contribution to this goal by providing a quick technique of detecting, classifying and logging signs. This strategy aids in the accurate and consistent development of the inventory. Once this is completed, human operators will have an easier time detecting distorted or obstructed signs.

Road and traffic sign recognition is a subject of study that can be used to help construct an inventory system (which does not require real-time recognition) or to help develop an in-car advising system (which does require real-time recognition). Both road sign inventory and road sign identification deal with traffic signs, face comparable issues and rely on automatic detection and recognition.

In principle, a road and traffic sign recognition system could be developed as part of an Intelligent Transport Systems (ITS) in figure 2.1 that continuously monitors the driver, the vehicle and the road in order, for example, to inform the driver in real time about upcoming navigational decision points and potentially hazardous traffic situations [5].

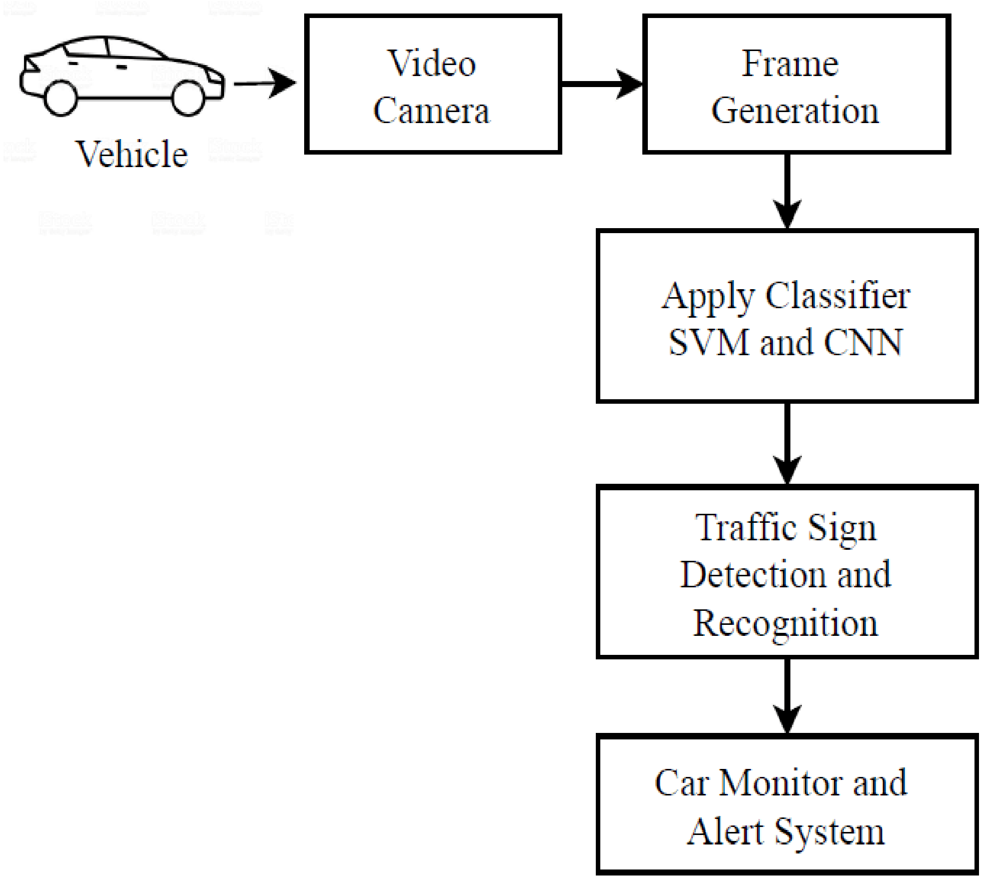


Figure 2.1: Block diagram of ITS [6]

Chapter 3

Problem Statement

Road signs can be located in a variety of situations, such as old, damaged, disoriented, etc., in addition to the complex environment of the roads and the surroundings. As a result, the detection and recognition of these signs may encounter one or more of the following challenges:

- Due to prolonged exposure to sunlight and the interaction of the paint with the air, the colour of the sign fades with time as shown in Figure 3.1.

Figure 3.1: Faded Signs
Source: Self Taken

- ❖ As seen in Figure 3.2, meteorological factors including fog, rain, clouds, and snow have an impact on visibility.

Figure 3.2: Bad weather conditions (Rain and Fog)
Source: Self Taken

- ❖ Local light differences like as the direction of the light, the intensity of the light depending on the time of day and season, and the shadows cast by other objects can all have an impact on visibility. As shown in Figure 3.3.
- ❖ Colour information is extremely sensitive to changes in lighting circumstances, including shadows, clouds, and sunlight [7, 8, 9]. Illuminate colour daylight, illumination geometry and viewing geometry can all have an impact on it.

Figure 3.3: Bad lighting geometry
Source: Self Taken

❖ The presence of impediments such as trees, buildings, automobiles, pedestrians, or even signs that block other signs as shown in Figure 3.4.

Figure 3.4: The presence of obstacles in the scene
Source: Self Taken

❖ The existence of things in the area under examination that are the same colour as or have a comparable shape to the road signs, such as buildings or cars [8, 10]. They may resemble the road sign in terms of colour, shape, or even both. In the first scenario, a fence that matches the hue of the road sign is shown in Figure 3.5. In the second instance, the post box is situated adjacent to the road sign and is the same size, shape and colour as the signs.

Figure 3.5: Similar objects in the scene or similar background colour
Source: Self Taken

- ❖ Any sort of obstruction, including other signs, can cause signs to become disoriented, damaged, or obscured. Two distinct damaged signs are depicted in the signs in Figure 3.6. The red rim on the one on the left is damaged, whilst the one on the right is really old, rusty, damaged and the colour has faded with age.

Figure 3.6: Damaged Signs

Source: Self Taken

- Motion blur and vehicle vibration are common problems with the captured image [11]. Because the recognition mechanism is unaware of the car movements above a certain threshold, this motion blur cannot be predicted. If an object is moving continuously and unmodified, it is feasible to predict how it will move in the future. Two distorted images are displayed in Figure 3.7. The image on the left is obviously blurred by motion. The sign is visible in the image on the right, but the background has a noticeable motion blur.

Figure 3.7: Motion blur problem
Source: Self Taken

The aforementioned potential issues can be used to draw the conclusion that it is crucial to develop algorithms for the detection and recognition of traffic and road signs that have high colour segmentation robustness, high noise and brightness insensitivity and should be invariant to geometrical effects like translation, in-plane and out-of-plane rotations and scaling changes in the image [12, 13].

Chapter 4

Objectives of The Research

The main objective is to create a system that can be used to catalogue traffic signs. By automatically identifying and categorising one or more traffic signs from a complicated picture when they are photographed by a camera mounted on a moving vehicle, this technology can help local or national authorities with the chore of maintaining and updating its road and traffic signs.

Finding the perfect colour combination in the scene, combining it with the appropriate form and placing one colour inside the convex hull of another colour, is the basic technique. If a candidate is discovered, the system attempts to categorise the object using the rim-pictogram combination and displays the categorization outcome.

The objectives are thus:

1. To understand how the characteristics of traffic and road signs may influence image processing for the recognition challenge.
2. Using invariant shape measures, create a recogniser that is resistant to in-plane transformations including translation, rotation and scaling.
3. To determine the ideal technique for feature extraction from traffic signs.
4. To create a classification system for road signs that is adequate.
5. To assess the methods' performance for robustness under various weather, illumination geometry and sign situations.

Chapter 5

Organisation of Thesis

- The first section of this thesis covers the goals and objectives as well as how it relates to the topic of intelligent transportation systems.

- The second section explains the characteristics, classifications and appropriate colours and designs of traffic and road signs. It focuses on Indian traffic and road signs as well as distinctions between Indian and international standards.

- The third section illustrates why automating the work of categorising road sign recognition is crucial for highway authorities and why it is challenging for academics to do. It outlines the difficulties that must be overcome when taking pictures outside and it ends by stating the need for a workable scientific solution.

- The fourth section presents a review of the literature and earlier research. The material covered includes papers, technical reports and internet resources that were gathered for review. The review contains a study and analysis of algorithms for colour segmentation, shape identification and recognition, classification of traffic signs and classification of pictograms. It also discusses the many methods used to complete the computer vision problem of recognising traffic signs.

This section examines colour, including its stability and qualities under various lighting and geometric situations. As the primary source of data for colour segmentation, it focuses on hue.

- The traffic sign recognition system's construction technique is covered in the fifth section. The image collection and image database are shown first and then the colour segmentation algorithms created for this study are presented. The chapter includes information on the recognition algorithm that was created to recognise traffic signs in a scenario. The chapter concludes by describing the categorization technique used to recognise traffic signs. The evaluation involves training using various features, CNN architectures, hyperparameters and training methodologies. The quest for ideal values is described in the end. The CNN method was used in experiments to examine different features, architectures, hyperparameters, and training approaches. The goal of the CNN-based traffic sign detection and identification system is still to find the best values and configurations.

- The final section provides specifics on the experimental work done to assess the different algorithms proposed in the previous chapter. Different weather, lighting, and sign conditions were used to test colour segmentation algorithms. The data was labelled after pre-processing and fed into a convolution neural network (CNN). It classifies the testing data after the training phase.

Chapter 6

Literature Survey

Road sign identification can be divided into two basic phases: detection and recognition. Three research groups are grouped under "detection." The first group of researchers thinks that the colours of traffic signs provide crucial information for their identification and classification. The second group thinks that simply the shape of traffic signs can be used to detect them, while the third group thinks that colour and shape work together to detect any road signs. The photographs from actual traffic situations, which are comparable to the ones gathered during this research, were used in all of the publications that were reviewed.

The methods used to recognise traffic signs vary from one author to the next. There are numerous methods used to address this problem.

- Dr. A. Sivasangari et al. recommend in their article splitting the field procedures under review into two rule groups: possibility based, form-based systems. The proposed system is exhaustively apportioned into, data planning, data gathering and getting ready and testing. System uses variety of picture planning strategies to improve the image quality and to oust on-illuminating pixel and recognizing edges. Feature extractors are used to find the features of picture. Moved AI figuring Convolutional Neural Networks (CNN) is used to gather the differing traffic sign pictures reliant on their features by using the progressing camera [14].

- Achmad Zulfajri Syaharuddin et al. proposed to measure the level of precision in monitoring traffic signs (detection speed of 4-6 frames per second) from video recording (single camera) using the Faster Region based Convolutional Neural Network (Faster R-CNN) algorithm. The traffic sign detection system uses the Faster R-CNN algorithm with Inception v2 model which is implemented in the TensorFlow API framework. The Faster R-CNN consists of 2 different modules. The first module is a deep convolutional neural network which functions to build the area to be detected, which is called the Regional Proposal Network (RPN), and the second module is the Fast R-CNN detector which functions to use the previously proposed area. This system is one unit, a detection network based on the results of the manufacture and testing of a traffic sign detection system based on the Faster R-CNN method, so it can be shown that there is no difference in the results of detection of traffic signs in day and night conditions. Where the precision testing for detection of traffic signs during the day and at night is 100% [15].

- Raghunandan et al. proposed detection accuracy, RGB Euclidian Threshold 'T' in Target Detection, Y, Cb and Cr in Skin Detection have been simulated and implemented to improve the efficiency of the algorithmes for video surveillance applications [16].

- Kai Li et al. proposed an algorithm which can quickly locate and extracting the sign area and have more accurately identify traffic indication symbol by color detection based on HSI model and shape detection based on Hough transform [17].

- Marco Costa et al. proposes a cut-off-based methodology use First-fixation partition was straight related to speed and fixation range. Road signs were taken a gander at a ton closer partition than their detectable quality detachment. In a second report a staircase technique was used to test the acquaintance time edge that led with 75% precision in road sign ID. The utmost was 35 ms, showing that short fixations to a road sign could incite a correct ID [18].

- Gómez-Moreno et al. proposed the quantitative comparison of many methods of segmentation (including modern ones) that have been used effectively in identifying traffic signs. It is possible to classify the presented methods into colour-space thresholding, edge detection, and chromatic / achromatic decomposition. The segmentation method of the support vector machine (SVM) and speed enhancement using a lookup table (LUT) have also been tested [19].

- P. Garg, et al. Comparison of YOLOv2, Faster R-CNN and SSD methods to detect road signs. The test method uses the German Traffic Sign Dataset Benchmark (GTSDB) which consists of 7832 training images and 530 testing images with various conditions. The results obtained are the level of precision of the SSD is 87.30%, Faster R-CNN is 94% and YOLOv2 is 97.90% [20].

- L. Estevez et al. suggested an algorithm capable of recognising the Stop, Yield and Do-Not-Enter traffic warning signs. It consists of six modules: colour segmentation, edge localisation, RGB differencing, edge detection, histogram extraction and classification. Colour segmentation is only used to localise red edge areas; the segmentation is performed sparsely and interpixel segmentation distance is determined [21].

- Yuille et al. designed a sign finder system to help visually impaired people. The author assumed that signs consist of two colours (one for the sign and another for the text) and sign boundaries are stereotyped (rectangle, hexagonal). Based on a set of tests to determine seeds, a region growing algorithm is used to detect hypothesis regions [22].

- Yabuki et al. proposed a method to detect the road sign by using the colour distribution of the sign in XYZ colour space. They constructed a colour similaritymap from the colour distribution, which is then incorporated into the image function of an active net model. It is possible to extract the road sign when it is wrapped up in an active net [23].

- Fang et al. calculated the hue value of the HSI colour space for every pixel and the similarity between this hue and the stored hue values of particular colours in road signs is calculated. The maximum degree of similarity is then considered. This result is fed into a perceptual analyser to specify the colour of the sign [24].

- Bénallal and Meunier developed a computer vision system which is embedded in a car and capable of identifying road signs. Many experiments were carried out with several road signs to study the stability of colours under different illumination conditions. Segmentation is achieved by the RGB colour space. It is shown that differences between red and green and blue components respectively are high and could be used with an appropriate threshold for segmentation [25].

- H. Gómez-Moreno et al. proposed a method to Goal assessment of division calculations for traffic sign acknowledgment [26].

- S. Visalini et al. proposed that Traffic Sign Recognition Using Convolution Neural Networks, International Journal of Innovative Research in Computer and Communication Engineering [27].

- Siva Krishna et al. proposed that Efficient Traffic Signboard Recognition System Using Convolutional Networks, International Symposium on Signal Processing and Intelligent Recognition Systems: Advances in Signal Processing and Intelligent Recognition Systems from pp 198-207 [28].

- Jingwei Cao et al. proposed that Improved Traffic Sign Detection and Recognition Algorithm for Intelligent Vehicles [29].

- Safat B. Wali et al. proposed that Vision-Based Traffic Sign Detection and Recognition Systems: Current Trends and Challenges [30].

- Adonis Santos et al. proposed that Real-Time Traffic Sign Detection and Recognition System for Assistive Driving [31].

Chapter 7
Background Study

The road and traffic signs that use a visual or symbolic language regarding the roads in front of the motorist are those that are taken into consideration in this thesis. In this thesis, the phrases are used interchangeably and they may also be used to refer to "road traffic signs" in other contexts. They give the motorist bits of knowledge that make driving convenient and safe. The direction sign, which shows the approaching directions for travelling to named towns or on numbered routes not symbolically but primarily by text, is a sort of sign that is NOT examined in this thesis.

In order to ensure regular updating and maintenance, road and traffic signs must be installed correctly in the required areas and an inventory is preferable. Meetings with the highway authorities in Sweden and Scotland indicated the lack of but necessity for a traffic sign inventory. By offering a quick technique of detecting, classifying, and logging signs, an automatic method of detecting and recognising traffic signs can significantly help achieve this goal. This process aids in the accurate and reliable development of the inventory. Once this is completed, it will be simpler for human operators to detect deformed or obscured signs [42].

The study of road and traffic sign recognition can be used to support the creation of an inventory system (for which real-time recognition is not necessary) or a vehicle advisory system (for which real-time recognition is required). Since traffic signs are involved, both road sign inventory and road sign recognition use automatic detection and recognition.

1.1 Traffic Signs

To control, warn, direct, or inform other road users, traffic signs, lights, and other traffic devices are utilised. With the smooth and predictable passage of all traffic, both vehicle and pedestrian, they contribute to the achievement of an acceptable level of road traffic quality and promote safety [42].

Because their shapes and colours may be easily distinguished from their surroundings, road and traffic signs are made to be easily seen by drivers. There are a variety of characteristics that distinguish road and traffic signs from their surroundings:

- Strict guidelines are followed when designing, producing and installing road signs.
- They are created in fixed 2-D shapes like rectangles, circles, triangles and octagons.
- The signs' colours are intended to contrast with the surroundings so that drivers can clearly recognise them.
- The sign category regulates the colours.
- The sign's text is one colour, while the remaining portions are another colour.
- The colour of the paint used to cover the sign should be matched to a certain visible spectrum wavelength.
- The signs are placed in well-defined areas relative to the road so that the driver can roughly predict where they will be.
- They might have a pictograph, a line of text or both.
- The use of set text fonts and character heights on traffic signs distinguishes them in every country.
- They may be partially obscured, distorted, damaged or grouped with other signs in a group, among other situations.

7.1.1 Mandatory Signs:

In order to control and guarantee the safe passage of automobiles and pedestrians on Indian roadways, mandatory traffic signs are essential. These signs are intended to express precise instructions and directives that all users of the road must heed in order to preserve law and order, avert accidents and guarantee a free flow of traffic. The requirements for legally required traffic signs in India are set and enforced by the Indian Road Congress (IRC) and the Ministry of Road Transport and Highways (MoRTH).

7.1.2 Warning Signs:

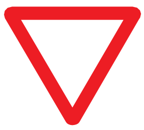
These signs are intended to warn motorists of potential dangers, impediments, or modifications in the state of the road up ahead. These signs assist drivers make educated judgements and take the required precautions by efficiently and rapidly communicating crucial information. This lowers the possibility of accidents and maintains traffic flow.

7.1.3 Prohibitory signs:

These signs notify motorists and pedestrians of important rules and information, which helps to reduce accidents and maintain traffic flow. Even when there are linguistic hurdles, standardised symbols and colours are used in the design of prohibitive signs. These warning signs point out prohibited behaviour and disobeying them can result in penalties and fines.

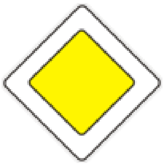
Here are some important traffic signs commonly found on Indian roads:

Table 7.1: Traffic sign names and their meanings

Traffic Sign	Name of Traffic Sign	Meaning
	Stop Sign (Mandatory Stop)	The Octagon is only used for STOP signs.
	Give Way Sign	Only use an equilateral triangle with the point down for YIELD signals.
	No Entry Sign	The circle is prohibiting vehicles from entering a specific area or road.
	One Way Sign	The arrow on the sign indicates the only direction in which traffic is moving.
	No Overtaking Sign	Stay behind in the no passing zone.
	No Overtaking Trucks	To make roads safer, maintain a safe distance.
	No Parking	It is not permitted to use the specified area for parking or stopping.

Traffic Sign	Name of Traffic Sign	Meaning
	Speed Limit 20KM/h	Vehicles with the allowed maximum speed limit 20.
	Speed Limit 30KM/h	Vehicles with the allowed maximum speed limit 30.
	Speed Limit 50KM/h	Vehicles with the allowed maximum speed limit 50.
	Speed Limit 60KM/h	Vehicles with the allowed maximum speed limit 60.
	Speed Limit 70KM/h	Vehicles with the allowed maximum speed limit 70.
	Speed Limit 80KM/h	Vehicles with the allowed maximum speed limit 80.
	Speed Limit 80KM/h	Vehicles with the allowed maximum speed limit 80.

Traffic Sign	Name of Traffic Sign	Meaning
	Speed Limit 100KM/h	Vehicles with the allowed maximum speed limit 100.
	Speed Limit 120KM/h	Vehicles with the allowed maximum speed limit 120.
	Crosswalk	Triangular sign that directs people to cross the street safely.
	Stagged Crosswords Sign	When your path crosses obstacles, move forward patiently and cautiously.
	No Lorries	Truck or lorry access is prohibited as indicated by the circular sign.
	Prohibiting Travel	Limitations on vehicle movement past this point are indicated.

Traffic Sign	Name of Traffic Sign	Meaning
	End of Major Road	This sign denotes the end of a prominent roadway, frequently, the changeover to a less important or developed route.
	Warning Sign	Drivers are warned by the triangular sign of potential dangers or potential changes in the state of the road.
	Curve to Left	There will soon be a left-turning bend in the road, and a sign warns to slow down and proceed cautiously.
	Curve to Right	There will soon be a right-turning bend in the road and a sign warns to slow down and proceed cautiously.
	Multiple Turns	Take Precautions for Complex Intersection.

Traffic Sign	Name of Traffic Sign	Meaning
	Speed Restrictions	Specifies particular speed limitations.
	Car Skid Warning	Warns drivers about the possibility of slick road conditions
	Road Narrows on Right	Shows that the right side of the road will get narrower in the future.
	Warning Sign about a Roadwork	Expect Delays and Lane Closures Due to Upcoming Roadworks
	Traffic Light Regulation	Utilises red, yellow, and green signals at junctions to regulate traffic and pedestrian crossings.
	Children	Observe children and move slowly.

Traffic Sign	Name of Traffic Sign	Meaning
	Intersection With a Cycle Path	Shows the presence of a crossover between a road and a designated bike lane.
	Snow Ahead Warning Sign	Indicating the presence of snowy or slippery conditions on the upcoming road.
	Wild Animals	Watch for Animals on Road.
	No Speed Limit	Shows that there is no maximum speed limit.
	Turn Right Sign	Shows that right turns are permitted for vehicles.
	Turn Left Sign	Shows that left turns are permitted for vehicles.
	Straight or Turn Right	Shows that cars may choose to move forward or make a right turn.

Traffic Sign	Name of Traffic Sign	Meaning
	Straight or Turn Left	Shows that cars may choose to move forward or make a left turn.
	Keep Right	Indicates that drivers should stay on the right side of the road.
	Keep Left	indicates that drivers should stay on the left side of the road.
	Roundabout Arrow	Shows the direction that cars should be driving in when approaching a circular intersection.
	End of Overtaking	Identifies a stretch of road when passing other cars is acceptable.
	End of Truck Overtaking	Shows the direction that No Passing for Trucks.

7.2. Colour image to grayscale conversion:

Converting RGB to grayscale is based on light perception and the human visual system. In the RGB colour model (red, green, blue), colours are represented as combinations of these three primary colours. Each pixel in an RGB image contains three colour channels with intensity values ranging from 0 to 255 for each colour (8 bits per channel), indicating the amount of red, green and blue light contained in that pixel. However, grayscale images are represented in only one channel, where each pixel has a single intensity value representing the brightness of the pixel. The grayscale colour space ranges from 0 (black) to 255 (white), with values in between representing different shades of grey [38].

The theoretical concept of RGB grayscale conversion involves assigning an appropriate intensity value to each pixel of an RGB image that represents its brightness.

There are several methods for this conversion, but one common approach is to use the weighted sum of the three colour channels to calculate the grayscale intensity. One popular formula for converting RGB pixels to grayscale is the lighting method, which takes into account the human perception of the brightness of different colours. The formula is:

$$\text{Gray} = 0.299 * \text{Red} + 0.587 * \text{Green} + 0.114 * \text{Blue}$$

The coefficients 0.299, 0.587 and 0.114 represent the relative contribution of red, green and blue to the total light. These coefficients are derived from the sensitivity of the human eye to different colours.

The green channel is most emphasized because our eyes are most sensitive to green light, followed by red and blue.

After applying this formula to each pixel of an RGB image, you get a grayscale image where each pixel has a single intensity value representing its brightness.

This process effectively removes the colour information from the original image and retains only the light information, leaving the image in black and white.

Figure 7.1: RGB to GRAY conversion [34]

7.3 Convolutional Neural Network (CNN):

A deep learning network design known as a convolutional neural network (CNN) learns directly from data. CNNs are incredibly helpful for identifying patterns in images that represent objects. For categorising non-image data, such as audio, time series and signal data, they can be highly useful [35].

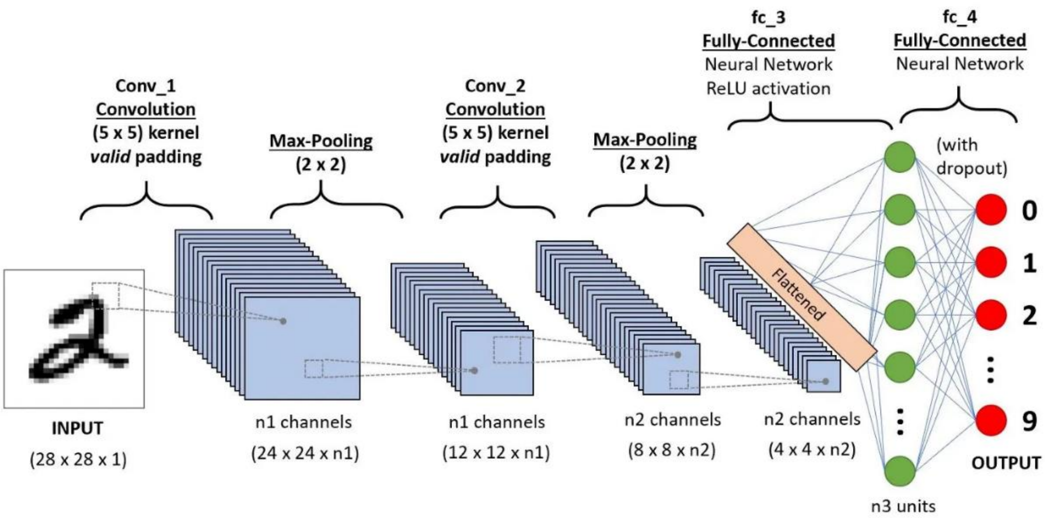


Figure 7.2: CNN Architecture [35]

Basic architecture of CNN – The architecture of CNN is showing here in figure 7.2. Architecture is explaining below.

- **Input image –**

The input RGB image consists of three-color planes – Red, Green and Blue as given in figure 7.3. It consists of pixel value at every point. The image is represented by height * width * no of channel. Height and width are number of pixels along height and width respectively.

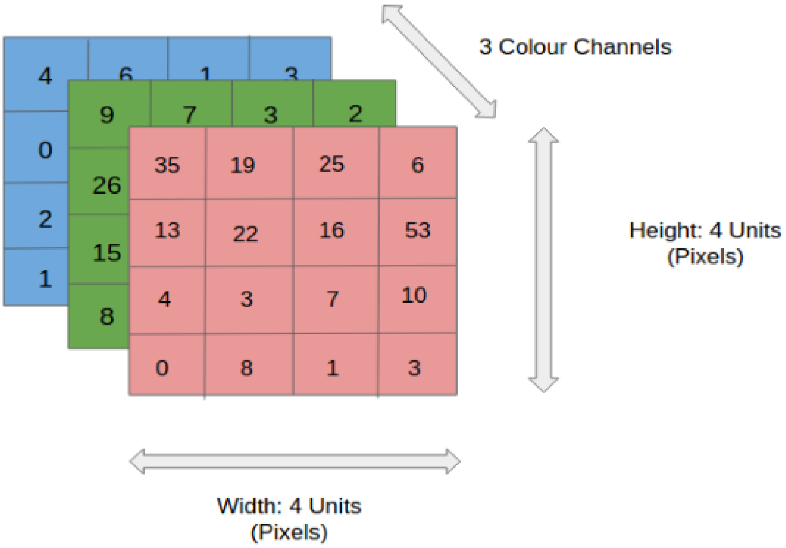


Figure 7.3: Input image [36]

- **Convolution layer-**

It is the core building block of a CNN because majority of calculation is done in this layer. It requires input image, feature map and filter. It has filter or kernel, which moves across the matrix of the image as shown in figure 7.4 and check for a certain feature is present or not. This process is called convolution.

The feature detector is a 2-D array of weights. Generally, 3x3 matrix is used as a feature detector.

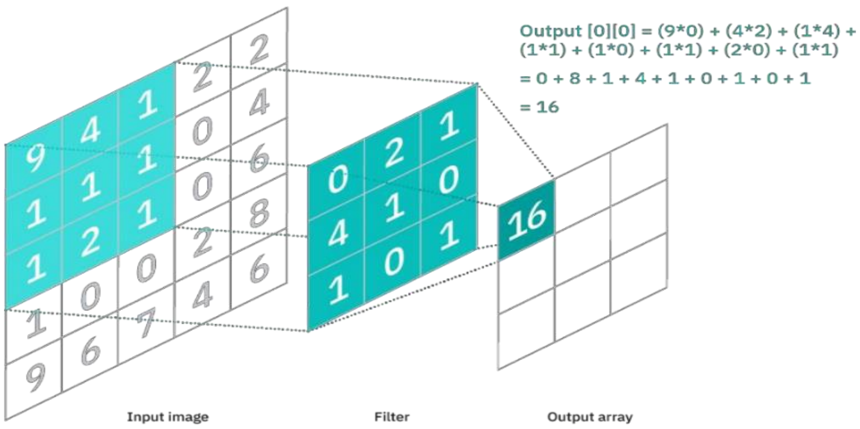


Figure 7.4: Convolution operation [35]

Convolution operation

It is observed from the picture that the filter is applied to portion of an image. Between the input pixel and filter a dot product is calculated. The result is sent to the output array. Then the filter shifts by a stride. This process repeats until the kernel sweep over the entire image. The result of dot product gives the final result in matrix form.

Kernel or Filter or Feature Detectors:

The kernel in a convolutional neural network is only a filter used to extract information from the images.

Formula = [i - k] + 1

i -> Size of input, K-> Size of kernel

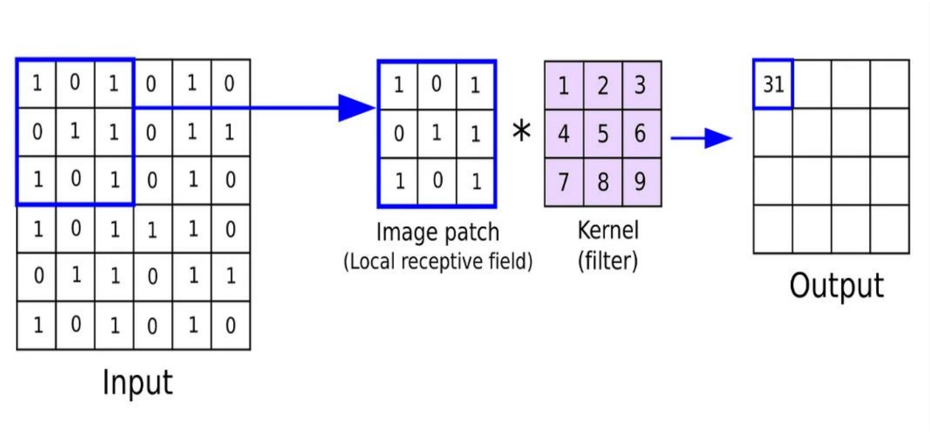


Figure 7.5: Kernal operation as a filter [35]

Stride- The neural network filter's step parameter called "stride" changes how much movement is present in an image or video. The steps will be taken one at a time as they already completed stride 1. If stride 2 is provided, the following 2 pixels will be skipped [35].

Formula = [i - k/s] + 1

i -> Size of input, K-> Size of kernel, S-> Stride

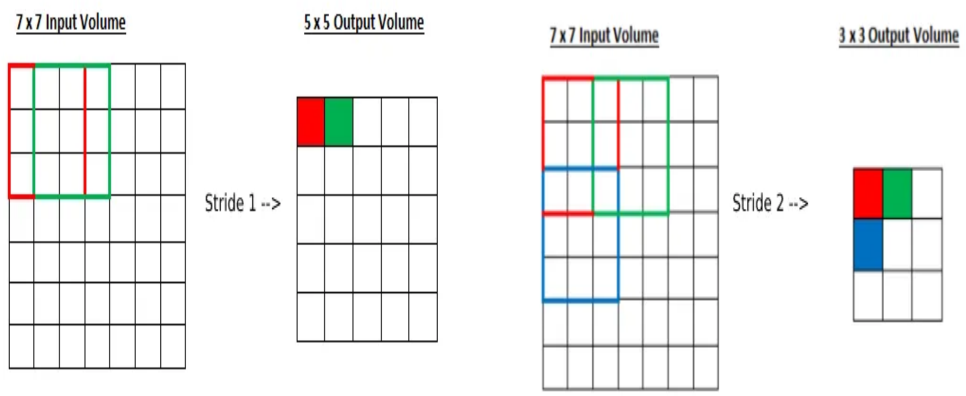


Figure 7.6: Stride [35]

Padding-

Convolutional neural networks (CNNs) can benefit from padding since it describes the number of pixels that are added to an image during processing by the CNN kernel. For instance, if the padding in a CNN is set to zero, then any additional pixels will have a value of 0. The size of the image will decrease when they use the filter or kernel to scan it. To keep the image's original size and extract some low-level information, they must avoid doing that. As a result, they will increase the image's size by a few pixels [35].

Formula= [i – k + 2p/s] + 1

i -> Size of input, K-> Size of kernel, S-> Stride, p->Padding

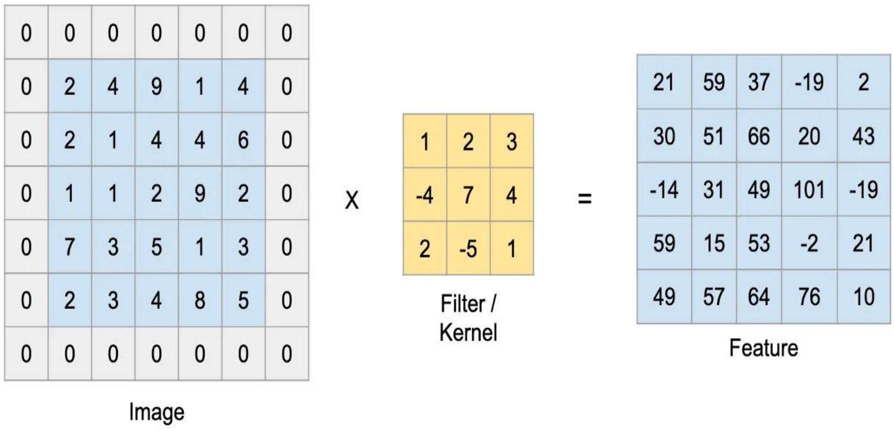


Figure 7.7: Kernal operation as a filter [35]

- **Flatten**

Flattening is the process of combining all of the 2-Dimensional arrays from the pooled feature maps into a single, long, continuous linear vector. The fully connected layer receives the flattened matrix as input to classify the image [35]. Figure 7.8 is showing this layer.

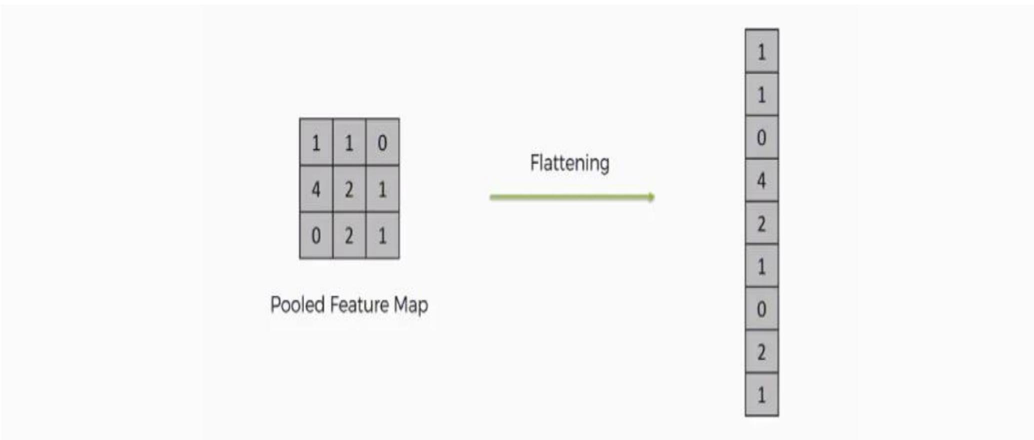


Figure 7.8: Flatten Layer [35]

- **Pooling layer-**

The dimensions of the feature map are reduced by the pooling layers. It sweeps a filter over the entire image matrix. An aggregation function is applied with the filter to get the output. There are 2 types of pooling:

i) Max pooling

ii) Average pooling

i. **Max Pooling-** As the filter moves over the image, it selects the max pixel value. Theseselected values make the output array as given in figure 7.9.

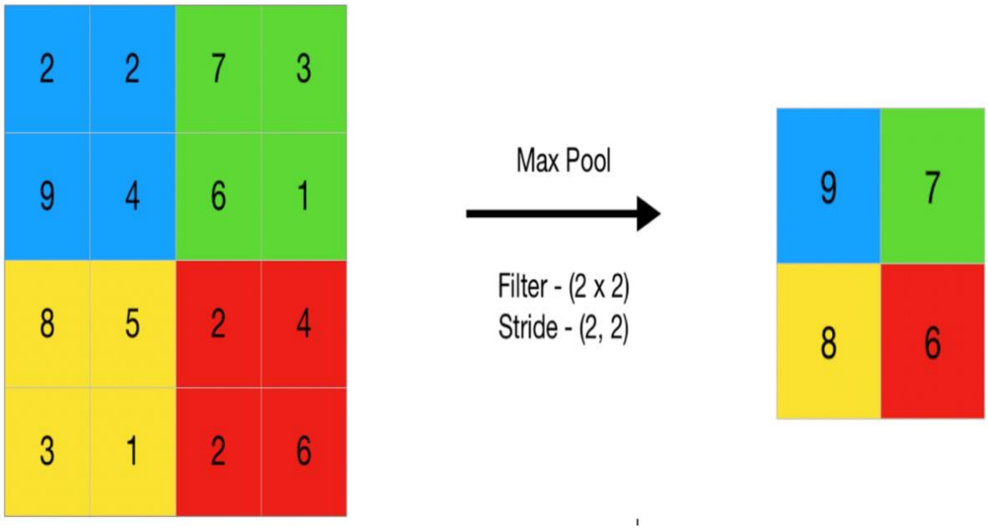


Figure 7.9: Max pooling operation [37]

ii) Average pooling

As the filter moves over the image, average value is calculated and sends to the output array as shown in figure 7.10.

Figure 7.10: Average pooling operation [37]

- **Dropout**

A Dropout layer is another common attribute of CNNs. The Dropout layer acts as a mask, erasing some neurons' contributions to the subsequent layer while leaving all others unaltered. Figure 7.11 is showing that layer.

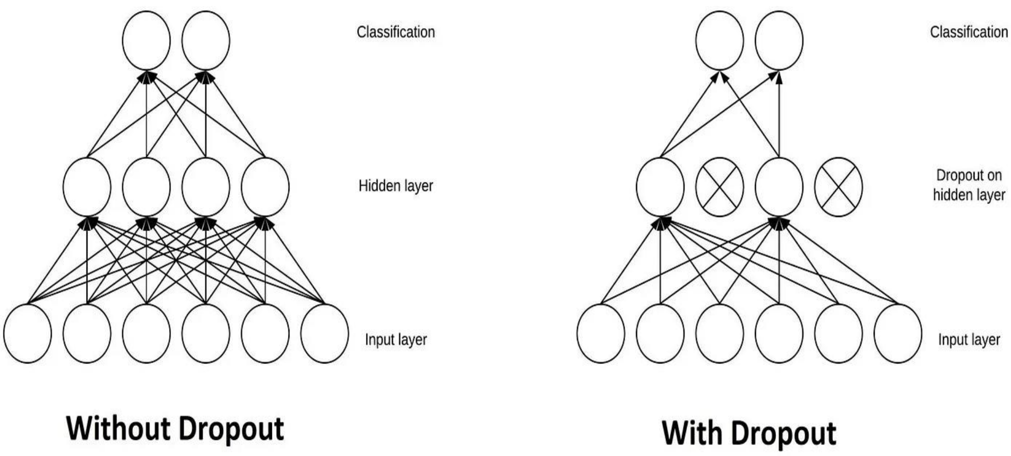


Figure 7.11: Dropout [35]

- **Activation Function**

A neuron's activation status

is determined by an activation function. As a result, it will determine whether or not the neuron's input to the network is crucial for prediction. The ReLU, Softmax, tanH and Sigmoid functions are a few examples of regularly used activation functions. Each of these operations has a particular use [35].

Sigmoid- For a CNN model binary classification.

tanH- The tanh function and the sigmoid function are quite similar. The symmetry around the origin is the only variation. In this instance, the possible values are between -1 and 1.

SoftMax- It is frequently employed as the final activation function of a neural network to normalize the output of a network to a probability distribution over expected output classes and is utilized in multinomial logistic regression.3

ReLU– ReLU denotes rectified linear unit and it is defined asan activation function given in Eq. [1].

$y = max\ (0, x)$ ------------- Equation 1

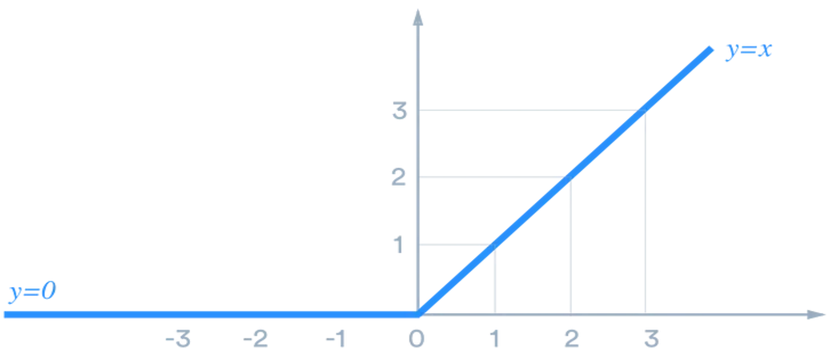


Figure 7.12: ReLU operation [39]

It gives zero output for all negative value and keeps the positive value same as given in figure 7.12.

- **Fully connected layer-**

In fully connected layer each neuron in one layer is attached to another neuron of another layer. When the flattened matrix goes through a fully connected layer it classifies the images.

- **Loss layer-**

It specifies how the predicted output is close to actual value. The loss function measures the difference between the predicted outputs and the ground-truth labels during training. For object detection, common loss functions include the Intersection over Union (IoU) loss and the Smooth L1 loss.

7.4 MATLAB:

The high-level programming language and interactive environment known as MATLAB, also known as "MATRIX LABORATORY," is largely used for numerical computing, data analysis, visualization and algorithm creation. It was created by MathWorks and was made available in the late 1980s. It has evolved into one of the most well-liked and frequently employed tools in a variety of disciplines, including engineering, physics, mathematics, finance and more [40].

Key features of MATLAB include:

1. Matrix-based Operations**:** Matrix-based Operations: MATLAB is extremely effective at managing massive quantities of numerical data because its core functionality is based on matrix

and array operations. This design decision enables the manipulation and concise display of mathematical formulas and algorithms.

2. Programming Environment: Programming Environment: MATLAB offers a user-interactive setting where users can create and run interpreted code. For swiftly reviewing data and testing algorithms, this capability is quite useful.

3. Rich Functionality: MATLAB has a huge library of in-built tools and functions for a variety of applications. For topics including signal processing, image processing, optimization, machine learning and more, these toolboxes offer specialized functions and tools.

4. Visualization: Data visualization is a crucial component of data analysis and MATLAB is a master at it. The numerous graphs, charts and animations that may be made using its robust plotting and visualization tools are provided.

5. Simulink: Simulink offers a graphical platform for modelling, simulating and analyzing dynamic systems. Simulink is an extension of MATLAB. It is extensively utilized in industries like communications, robotics and control systems.
6. App Development: With MATLAB, users may build their own graphical user interfaces (GUIs) and applications without having to have a deep understanding of programming. This function is useful for designing user-friendly user interfaces for sophisticated simulations or algorithms.

7. Integration: The integration enables users to use pre-existing code and libraries in their MATLAB processes by integrating with other programming languages like C, C++, Java and Python.

8. Community & Support: Because there is a sizable and active user base for MATLAB, there are a ton of online resources, discussion boards and documentation that can be used for problem-solving, education and idea-sharing.

Finally, MATLAB continues to be a potent tool for numerical computing, data analysis and algorithm development, especially in areas where its specialized toolboxes and features shine.

Machine Learning Model in MATLAB:

For creating machine learning models, MATLAB offers a complete platform. Different kinds of machine learning models can be created, trained and evaluated using built-in functions, toolboxes and the MATLAB programming language.

An overview of the general processes to build a machine learning model in MATLAB is provided below:

Data Preprocessing:

- Make use of MATLAB tools like "readtable" and "csvread" to load and import your dataset.
- By handling missing values, outliers and scaling features, you may clean up and preprocess your data.

Feature Extracting:

- Choose attributes that are pertinent to your model. Functions for feature extraction and selection are available in MATLAB.
- Utilize methods like PCA (Principal Component Analysis), LDA (Linear Discriminant Analysis), etc. to transform and design features to improve their representation.

Model Selection:

- Select the correct machine learning algorithm for your issue (such as clustering, classification, or regression).
- A variety of algorithms are supported by MATLAB's built-in functions and toolboxes.

Model Training:

- Using 'cvpartition' or manually, divide your dataset into training, validation and testing sets.
- The training data should be used to train your chosen model.
- Use methods like grid search or Bayesian optimization to adjust the hyperparameters.

Model Evaluation:

- Analyze the performance of your model on the validation and test sets.
- For measures like accuracy, precision, recall, F1-score, ROC curves, etc., MATLAB offers functions.

Chapter 8

Proposed Method

We proposed a traffic sign recognition system using CNN.

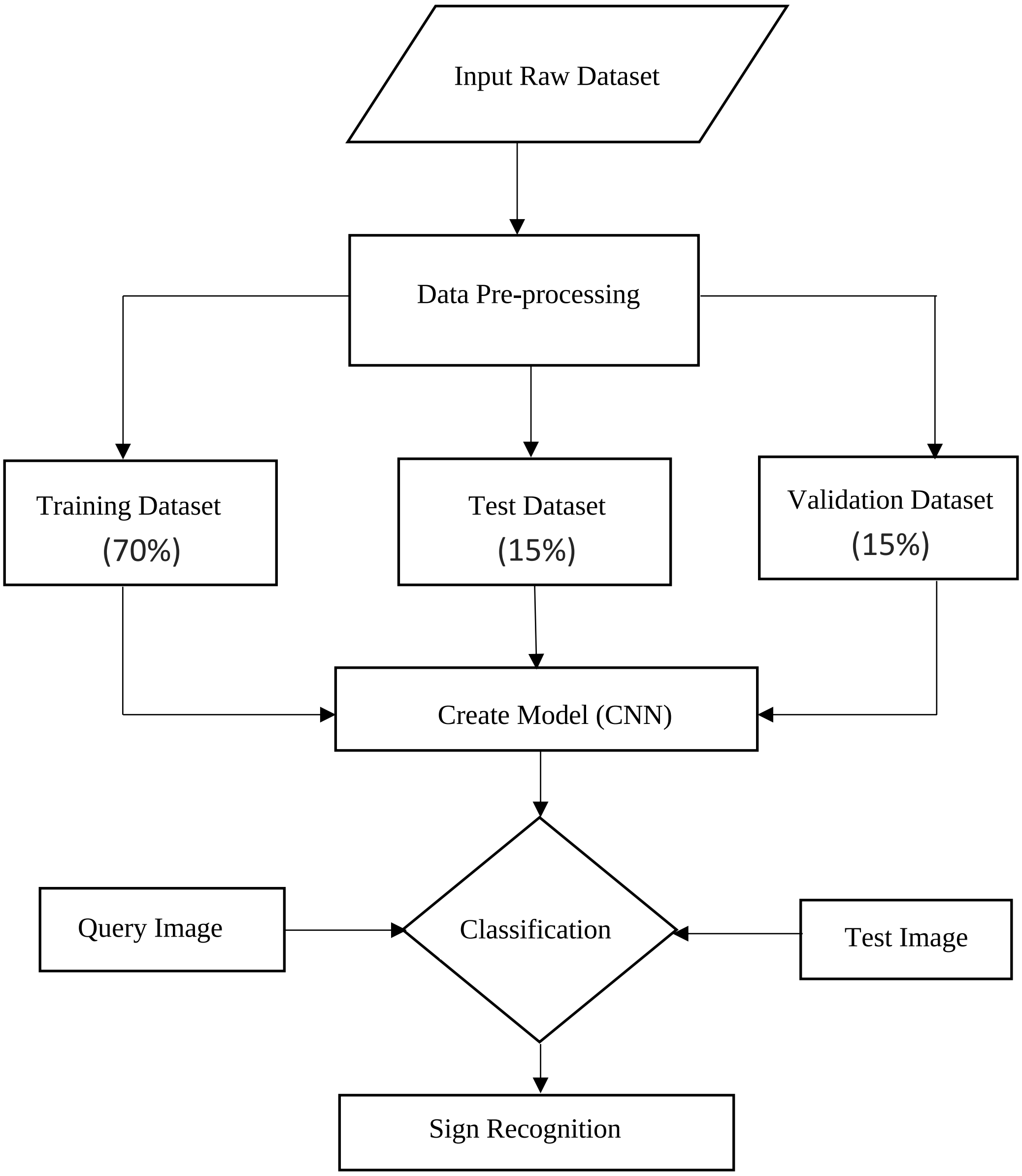


Figure 8.1: Proposed system architecture
Source: Self drawn using MS Word 2021

This figure 8.1 illustrates the flowchart of the system. At first, raw data is collected from various sources and processed into a dataset for the present system, as explained in Section 8.1. Then, the dataset goes through the preprocessing stage, where the data is resized and converted from RGB to grayscale, as described in Section 8.2. After that, the dataset consists of 39,209 images, which are divided into a training dataset (70% of the main dataset), a validation dataset (15% of the main dataset) and a testing dataset (15% of the main dataset). The dataset was then inputted into the CNN module for training, following the methodology outlined in Section 8.3 by the researcher conducting this study. Afterward, the researcher evaluates the efficiency of the system by classifying the images and predicting their labels using the test image dataset, which consists of 10,000 images. The researcher also uses real-life images as query images for the classification model and predicts their corresponding labels. After that, obtain the recognition sign as an output from the system.

Architecture is described here -

8.1 Dataset:

Input raw data are collected as dataset. For the recent research worker used known dataset [41]. It consists of 39209 data for training and 10000 data for test. Also, the present researcher considers some real traffic images for testing.

The training data are divided in 43 number of classes. Here are the details of those classes –

Table 8.1: Description of 43 classes

Class Name	Traffic Sign Details
0	20 Km/h Speed Limit
1	30 Km/h Speed Limit
2	50 Km/h Speed Limit
3	60 Km/h Speed Limit
4	70 Km/h Speed Limit
5	80 Km/h Speed Limit
6	80 Km/h Speed Limit
7	100 Km/h Speed Limit
8	120 Km/h Speed Limit
9	No Overtaking
10	No Overtaking Trucks
11	Stagged Crosswords Sign
12	End of Major Road
13	Give Way
14	Stop Sign
15	Prohibiting Travel
16	No Lorries
17	No Entry Sign
18	Warning Sign
19	Curve to Left
20	Curve to Right
21	Multiple Turns
22	Speed Restrictions
23	Car Skid Warning
24	Road Narrows on Right
25	Warning Sign about a Roadwork
26	Traffic Light Regulation
27	Crosswalk

Class Name	Traffic Sign Details
28	Children
29	Intersection With a Cycle Path
30	Snow Ahead Warning Sign
31	Wild Animals
32	No Speed Limit
33	Turn Right Sign
34	Turn Left Sign
35	One Way
36	Straight or Turn Right
37	Straight or Turn Left
38	Keep Right
39	Keep Left
40	Roundabout Arrow
41	End of Overtaking
42	End of Truck Overtaking

8.2 Data Preprocessing:

The present research proposes a method based on CNN. As this CNN model requires input images to be of size 32*32, resize the images to match this size. As this CNN model requires input images to be grayscale, it converts RGB images to grayscale. So, image resizing and converting RGB to grayscale are parts of data preprocessing. In Figure 8.2, the first image consists of raw data, while the second image consists of images after pre-processing.

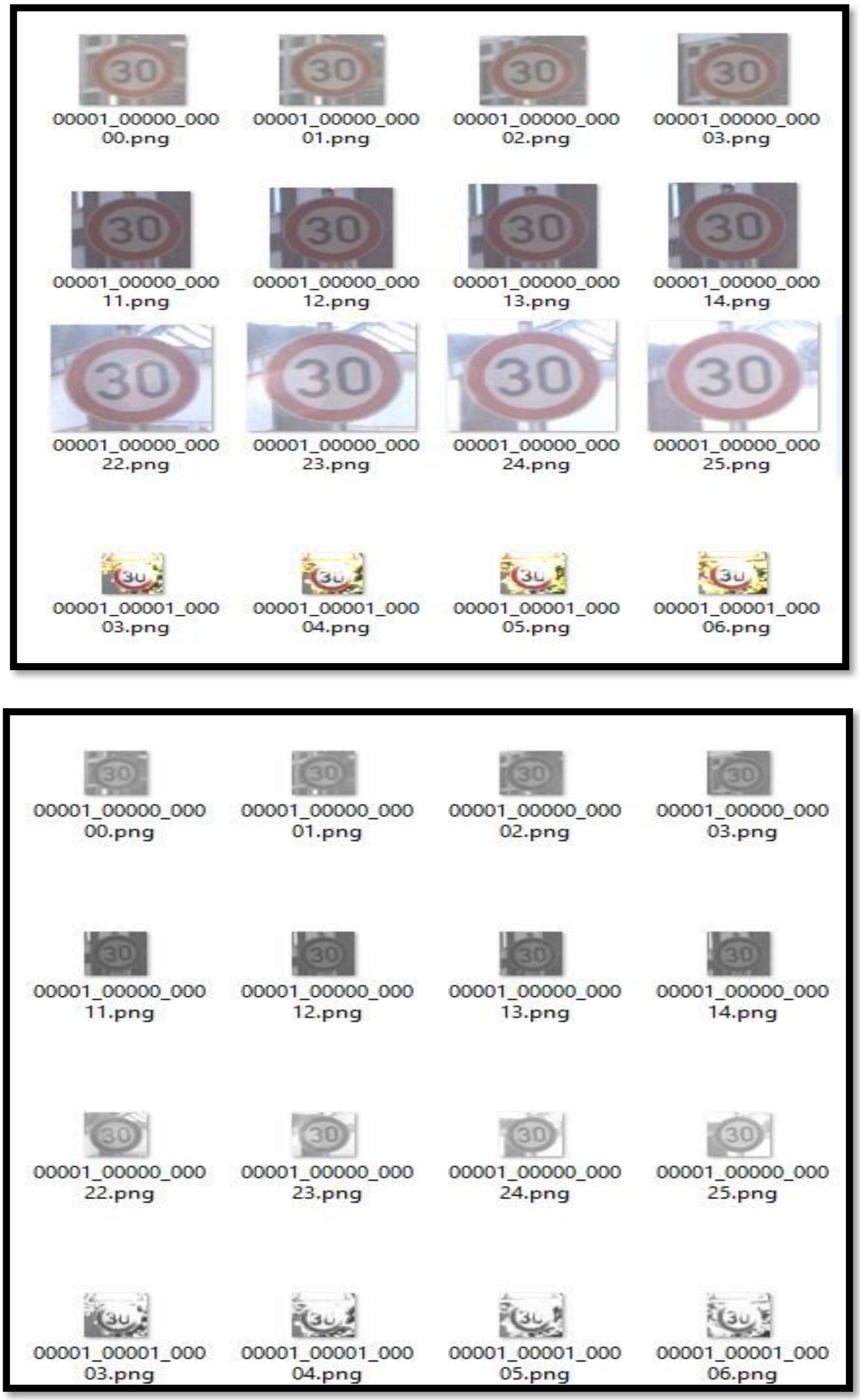


Figure 8.2: Data Pre-processing

Source: Data Pre-processing is obtained using MATLAB 2021a

8.3 Convolutional Neural Network (CNN):

The present researcher has defined a convolutional neural network architecture for this method. The present researcher has defined this in MATLAB using toolbox. This architecture seems to be designed for picture classification tasks, particularly with a dataset consisting of 43 classes.

Image input layer - This is the input layer, which expects grayscale images with a resolution of 32x32 pixels.

Convolution 2d layer - Three convolutional layers of different filter size are applied here in this method. First convolutional layer consists of 32 filters of (3*3) kernel size is applied here. Second convolutional layer consists of 64 filters and third convolutional layer consist of 128 filters.

Kernel size – Kernel size of (3*3) is applied for all convolutional layers.

Padding – Here the present researcher used ‘Zero Padding’ as the parameter in this methodology. The present researcher has considered padding parameter has the same spatial dimensions as the input.

Batch normalization layer – The present researcher has used the batch normalization layer that aids in the stabilization and acceleration of neural network training.

ReLU Layer – In the present method, the research worker has used ReLU as the activation function. The activation function of a rectified linear unit (ReLU) introduces nonlinearity into the network.

Max pooling 2d layer – The present researcher has designed the max-pooling layer that decreases the spatial dimensions of feature maps by a factor of two while keeping key characteristics. The present researcher has used pooling size as (2*2).

Stride – A stride of two is used in this method.

Fully connected layer – The present researcher has proposed the fully connected layer with 256 neurons. The last fully connected layer has 43 neurons, which correspond to the 43 categories in the classification job.

Dropout layer – The Dropout layer acts as a mask, erasing some neurons' contributions to the subsequent layer while leaving all others unaltered. The present researcher has proposed a dropout layer that randomly disables 50% of neuron activations during training. This helps to avoid overfitting.

SoftMax Layer - In the last section the present researcher has used the SoftMax activation layer that turns the raw output scores of the network into probability values that indicate the anticipated class probabilities.

Classification layer - The loss function for training and evaluation is specified in the output layer. It's set up for a multi-class classification operation in this system.

- ✓ In the present researcher's method Convolutional layers discover hierarchical features, max-pooling layers lower spatial dimensions and fully linked layers aggregate features for final classification. Better training and generalisation are aided by the batch normalisation and dropout layers.

- ✓ The training samples were split by the present researcher. This method is split into three parts and this splitting is based on randomly collected data from the train dataset. 70% of the data in the train dataset is used for training purposes, 15% for testing purposes and another 15% for validation purposes.

- ✓ The optimization algorithm that will be utilised during training is 'Adam'. Adam (Adaptive Moment Estimation) is a well-known optimisation algorithm that adjusts the learning rate during training.

- ✓ Mini-batch training is a typical technique that changes the network's weights in each iteration using a fraction of the training data. Here the present researcher used a batch size of 128. A batch size of 128 indicates that the network will process 128 samples before updating the weight.

- ✓ Max epochs specify the maximum number of training epochs, or how many times the full training dataset will be used for training. Here the present researcher used 4 epochs.

- ✓ Validation frequency parameter controls how frequently the network's performance on the validation dataset is evaluated. In this method they set this to 10. It's set to every 10 mini-batches in this scenario.

After training is done, they have done the classification with respect to the present researcher's test dataset, which contains 10000 images. From this, they got the predicted output. After that, the present researcher also did a query test. which provides an image to the classifier and gets the output of what type of traffic sign that is and the name of the sign, which can be feed to the alert system, or ITS.

Chapter 9

Experimental Results and Analysis

The performance of the proposed system is continuously evaluated throughout the test dataset which is consist of 10000 images. The present researcher's model is created and simulated in MATLAB software. Here seems to be the performance of this model as well as classification accuracy. The present researcher trained about 39209 images of different conditions in this model.

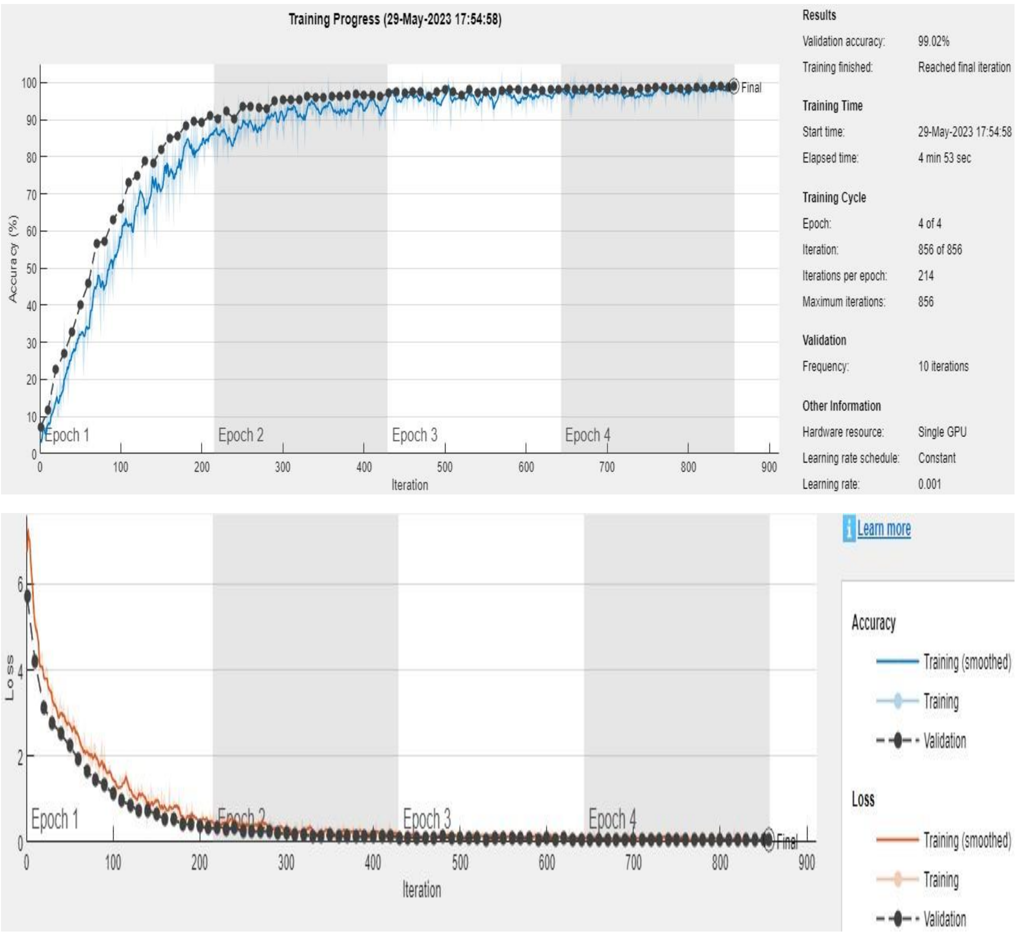


Figure 9.1: Present Model Performance
Source: Present Model Performance is obtained using MATLAB 2021a

In figure 9.1 the present researcher Here seems the performance of this model as present model. The model gives a 99.02% validation accuracy. Also see the maximum iterations are 856. Iterations per epoch are 214. Learning rate 0.001 on single GPU. Training time for 39202 images are 4 minute and 53 second. Here also seems to be an accuracy and loss curve with respect to iterations.

It can seem that the accuracy curve inclines in nature and the loss curve declines in nature as the training progresses. Finally reach the destination.

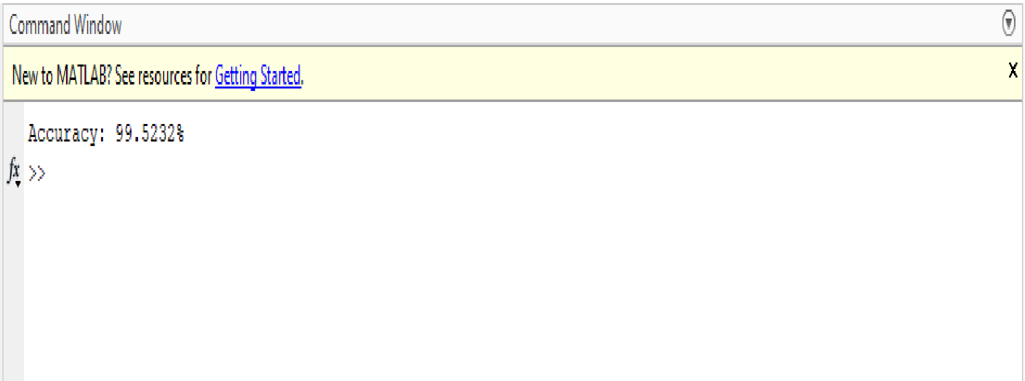


Figure 9.2: Accuracy of the proposed model

Source: Accuracy of the proposed model is obtained using MATLAB 2021a

In figure 9.2, the present researcher observes the accuracy of this system. This accuracy based on the test dataset of 10,000 and 6000 images. This is simulated in MATLAB.

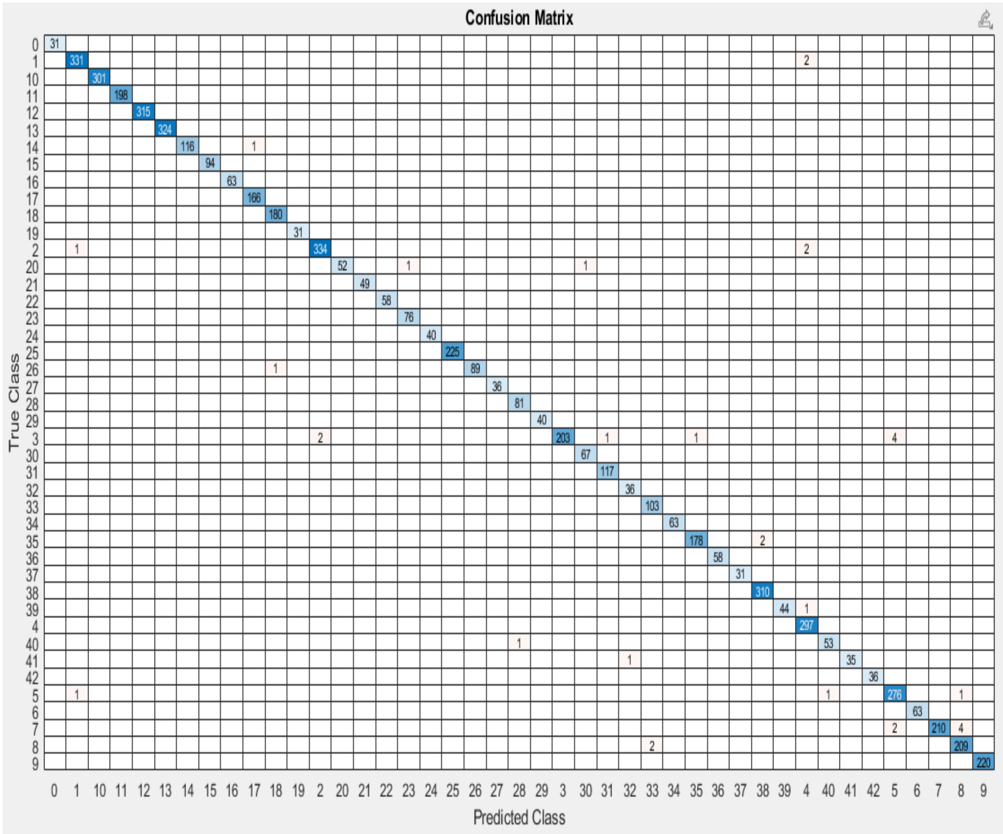


Figure 9.3: Confusion matrix of the proposed model

Source: Confusion matrix of the proposed model is obtained using MATLAB 2021a

This is the confusion matrix containing two parameters true labels and predicted labels. True labels are the actual classes of training dataset. Predicted labels are those labels which are predicted by the model. Labels are the classes of training dataset. The confusion matrix of this system is shown in the figure 9.3, the diagonal elements of this matrix are correctly classified. From there, the number of correctly classified images is identified. Other than diagonal elements, all are not correctly classified. There are 33 images that are not correctly classified as seen in the confusion matrix.

Figure 9.4: Predicted labels

Source: Predicted labels is obtained using MATLAB 2021a

In figure 9.4 the present researcher can see some of the predicted labels from testing image dataset. This is the output from the classifier and this is predicted by experiments. It seems that all images are correctly classified according to their respected labels.

Predicted labels are those labels which are predicted by the model. Labels are the classes of training dataset.


Figure 1
File Edit View Insert Tools Desktop Window Help
0
MOBILE:- 7557805910
20


Figure 1
File Edit View Insert Tools Desktop Window Help
19

Figure 9.5: Predicted labels from query image

Source: Predicted labels from query image is obtained using MATLAB 2021a

This is the hypothesis and it has been experimentally verified. In Figure 9.5, four images are shown displaying predicted labels for corresponding images (which are different from the present researcher's dataset). Label 0 indicates a 20 km/h speed limit. Label 19 means "Curve to the Left" Label 5 indicates an "80 km/h speed limit". So, it can be said that they have been classified correctly.

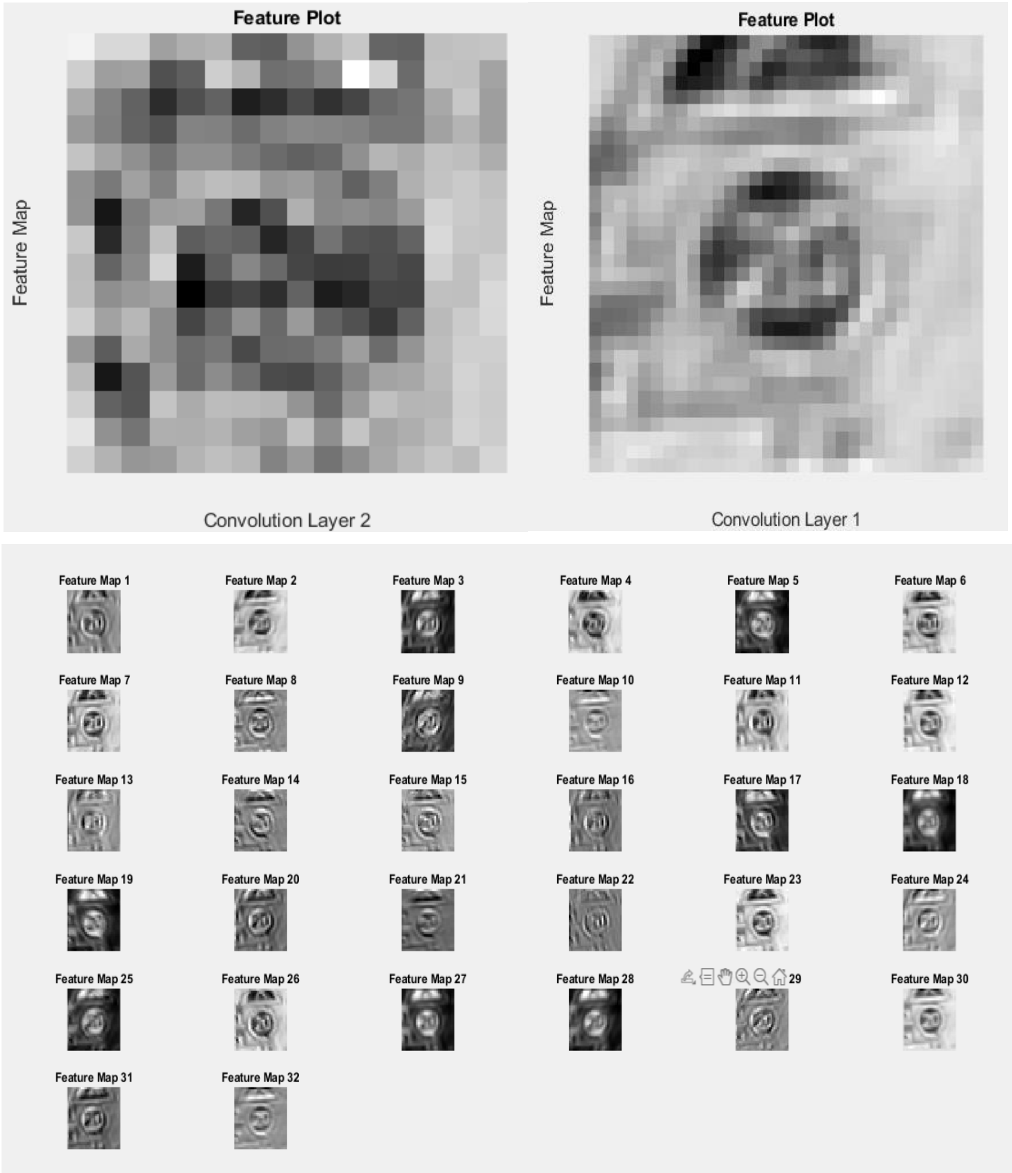


Figure 9.6: Feature Plot

Source: Feature Plot is obtained using MATLAB 2021a

In Figure 9.6, the present researcher can see the feature plot. Here are the feature plots for the first two convolutional layers, displaying the feature plot for one class. In the first layer, there are 32 filters. So, there are 32 feature maps for one class, which are shown in the figure.

Chapter 10

Comparative Analysis

The present researcher proposed a traffic sign recognition system based on CNN (Convolutional Neural Network). In this work, they used a large dataset of 39,202 images was used for training purposes. Additionally, the present proposed model was evaluated using 6000 and 10000 images for the test dataset. An accuracy of 99.52% has been achieved here, which is significantly higher compared to other proposed methods. Despite using a large dataset, it can be said that it has significantly improved the accuracy level compared to others, as shown in Table 10.1.

Table 10.1: Comparison Result

Method	**Accuracy**
Staircase technique [18]	75%
SSD [20]	87.30%
Faster R-CNN [20]	94%
YOLOv2 [20]	97.90%
CNN (Proposed Method by the present researcher)	**99.52%**

Chapter 11

Conclusion and Future Scope

11.1 Conclusion

This thesis designed, implemented and evaluated a system for recognizing traffic and road signs, which can assist in creating a road sign inventory. This method, which combines pattern recognition and computer vision techniques, was able to extract traffic signs from still photos of complex situations with unpredictable lighting conditions. The present researcher utilizes a deep learning model for traffic sign recognition. This research work has achieved a very high accuracy rate of 99.52%. This work also utilizes a large dataset for the analysis. A total of 39,209 images were used for training, and 10,000 images were used for testing in this research work. Compared to other models and datasets, this method yields highly challenging and impressive results.

11.2 Future Scope

The study started here could open up a number of new research directions. The following is a list of them:

- Object and Occluded Sign Recognition- Utilising features that are unaffected by clutter or partial occlusion will help fix this issue.
- Removal of Signs.
- Measures of Similarity for Sign Detection.
- Real Time Applications.
- New and improved model architectures can be explored as technology advances. This could include more advanced

convolutional neural networks (CNNs), attention mechanisms, or perhaps self-attention and transformer parts. These architectures may improve system's ability to recognise signs in difficult environments.

- Integrating different data sources, such as pictures, LIDAR and radar data, could result in more robust recognition algorithms. This method would allow systems to recognise traffic signs in a variety of weather and lighting circumstances.
- Domain adaptation approaches may aid traffic sign recognition models in performing effectively when travelling from one geographical location to another, taking into consideration differing sign designs, colours and even languages.
- An emerging subject is the study of how autonomous cars can communicate with pedestrians and other drivers via traffic signs or signals. This entails creating systems that "understand" and respond to human motions and signals.
- As traffic sign recognition technology becomes more common, there will be debates about its legal and ethical implications. How should these systems be controlled? What are the implications for driving behaviour and road safety?
- To increase the dependability and safety of autonomous and semi-autonomous cars, the future of traffic sign recognition will require refining existing models, developing new architectural paradigms, and solving real-world difficulties. This topic combines computer vision, machine learning, transportation engineering and human-computer interaction, making it a fascinating and developing field of study.

References

[1] A . Mogelmose, M. M. Trivedi and T. B. Moeslund, *"Vision-based traffic sign recognition and examination for canny driver help frameworks: Perspectives and overview*," IEEE Trans. Intel. Transp. Syst., vol. 13, no. 4, pp. 14841497, Dec.2012.

[2] H. Gómez-Moreno, S. Maldonado-Bascón, P. Gil-Jiménez and S. Lafuente-Arroyo, *"Goal assessment of division calculations for traffic sign acknowledgment,*" IEEE Trans. Intel. Transp. Syst., vol. 11, no. 4, pp. 917930, Dec.2010.

[3] S. Visalini, *"Traffic Sign Recognition Using Convolution Neural Networks,"* in IEEE Int. J. Innov. Res. Comput. Commun. Eng., vol. 5, no. 6, pp. 12345-12350, Jun. 2017.

[4] S. K. P. Mothukuri, R. Tejas, S. Patil, V. D. Shashidhar and G. Koolagudi, *"Efficient Traffic Signboard Recognition System Using Convolutional Networks,"* in Proceedings of the 5th International Symposium on Signal Processing and Intelligent Recognition Systems: Advances in Signal Processing and Intelligent Recognition Systems (SIRS 2019), pp. 198-207, Trivandrum, India, December 18–21, 2019.

[5] US Department of Transportation, *"Intelligent Transportation Systems", URL:* http://www.its.dot.gov/its_overview.html, 2006.

[6] https://www.researchgate.net/figure/Basic-block-diagram-for-TSRS_fig1_344399165

[7] C. Fang, C. Fuh, S. Chen, and P. Yen, *"A road sign recognition system based on dynamic visual model,"* presented at the 2003 IEEE Computer Society Conf. Computer Vision and Pattern Recognition, Madison, Wisconsin, 2003.

[8] J. Miura, T. Kanda, and Y. Shirai, *"An active vision system for real-time traffic sign recognition,"* presented at 2000 IEEE Intelligent Transportation Systems, Dearborn, MI, USA, 2000.

[9] S. Vitabile, G. Pollaccia, G. Pilato and F. Sorbello, *"Road sign Recognition usinga dynamic pixel aggregation technique in the HSV color space,"* presented at 11thInter. Conf. Image Analysis and Processing, Palermo, Italy, 2001.

[10] M. Blancard, "Road Sign Recognition: A study of Vision-based Decision Making for Road Environment Recognition," in Vision-based Vehicle Guidance, I. Masaki, Ed. Berlin, Germany: Springer-Verlag, 1992, pp. 162-172, April 1996.

[11] P. Paclik and J. Novovicova, *"Road sign classification without color information,"* presented at Sixth Annual Conf. of the Advanced School for Computing and Imaging, Lommel, Belgium, 2000.

[12] E. Perez and B. Javidi, *"Composite filter bank for road sign recognition,"* presented at 13th Annual Meeting IEEE Lasers and Electro-Optics Society, Rio Grande, Puerto Rico, 2000.

[13] D. Kang, N. Griswold and N. Kehtarnavaz, *"An invariant traffic sign recognition system based on sequential color processing and geometrical transformation,"* presented at IEEE Southwest Symposium on Image Analysis and Interpretation, Dallas, Texas, USA, 1994.

[14] Dr. A. Sivasangari, Dr. P. Ajitha, Dr. R.M. Gomathi , S. Nivetha, Pavithra *"Indian Traffic Sign Board Recognition and Driver Alert System Using CNN"* presented at IEEE Xplore, 2020.

[15] Achmad Zulfajri Syaharuddin, Zahir Zainuddin, Andani *"Multi-Pole Road Sign Detection Based on Faster Region-based Convolutional Neural Network (Faster R-CNN)"* presented at IEEE Xplore, 2021.

[16] A. Raghunandan, Mohana, P. Raghav and H. V. Ravish Aradhya *"Object Detection Algorithms for Video Surveillance Applications"* presented at IEEE,2018.

[17] Kai Li , Weiyao Lan **"***Traffic indication symbols recognition with shape context"* presented at IEEE, 2011.

[18] M. Costa, A. Simone, V. Vignali and C. Lantieri, *"Fixation distance and fixation duration to vertical road signs,"* in IEEE Transactions on Intelligent Transportation Systems, vol. 69, no. 1, pp. 48-57, 2018.

[19] H. Gomez-Moreno, S. Maldonado-Bascon, P. Gil-Jimenez and S. Lafuente-Arroyo, *"Goal Evaluation of Segmentation Algorithms for Traffic Sign Recognition,"* IEEE Transactions on Intelligent Transportation Systems, vol. 12, no. 1, pp. 158-167, January 2011. DOI: 10.1109/TITS.2010.2054084.

[20] P. Garg, D. R. Chowdhury and V. N. More, *"Traffic Sign Recognition and Classification Using YOLOv2, Faster RCNN and SSD,"* 2019, pp. 1–5.

[21] L. Estevez, N. Kehtarnavaz *"A real-time histographic approach to road sign recognition"*, IEEE,2022.

[22] A.L. Yuille; D. Snow; M. Nitzberg, *“Signfinder: using color to detect, localize and identify informational signs”*, IEEE, 2002.

[23] Takuya Ueta, Yasuaki Sumi, Noboru Yabuki, Susumu Matsumae, Yoshihiro Fukumoto, Takao Tsukut *“A Study on Contour Line and Internal Area Extraction Method by using the Self-Organization Map”*,IEEE,2007.

[24] C. Fang, S. Chen and C. Fuh, *"Road-sign detection and tracking," IEEE Trans. on Vehicular Technology*, vol. 52, pp. 1329-1341, 2003.

[25] Hasan Fleyeh, *“Shadow and Highlight Invariant Colour Segmentation Algorithm for Traffic Signs”*, IEEE, 2006.

[26] H. Gómez-Moreno, S. Maldonado-Bascón, P. Gil-Jiménez and S. Lafuente-Arroyo, *"Goal assessment of division calculations for trafc sign acknowledgment,"* IEEE Trans. Intell. Transp. Syst., vol. 11, no. 4, pp. 917930, Dec.2010.

[27] S. Visalini, *"Traffic Sign Recognition Using Convolution Neural Networks,"* in IEEE International Journal of Innovative Research in Computer and Communication Engineering, vol. 5, no. 6, pp. 12345-12350, June 2017.

[28] S. K. P. Mothukuri, R. Tejas, Soham Patil and V. Darshan Shashidhar G. Koolagudi, *"Efficient Traffic Signboard Recognition System Using Convolutional Networks,"* in Proceedings of the International Symposium on Signal Processing and Intelligent Recognition Systems: Advances in Signal Processing and Intelligent Recognition Systems, pp. 198-207, 2020.

[29] Jingwei Cao , Chuanxue Song , Silun Peng , Feng Xiao and Shixin Song*," Improved Traffic Sign Detection and Recognition Algorithm for Intelligent Vehicles",* Sensors 2019, 19, 4021;doi:10.3390/s9184021.

[30] Safat B. Wali , Majid A. Abdullah , Mahammad A. Hannan , Aini Hussain , Salina A. Samad , Pin J. Ker and Muhamad Bin Mansor*," Vision-Based Traffic Sign Detection and Recognition Systems: Current Trends and Challenges"*, Sensors 2019, 19, 2093; doi:10.3390/s19092093.

[31] Adonis Santos, Patricia Angela Abu, Carlos Oppus, Rosula Reyes*," Real-Time Traffic Sign Detection and Recognition System for Assistive Driving"*, Volume 5, Issue 4, Page No 600-611, 2020.

[32] https://www.news18.com/news/lifestyle/adding-kiwi-fruit-to-your-diet-has-many-benefits-heres-what-you-need-to-know-4264979.html

[33] https://medium.com/@draj0718/convolutional-neural-networks-cnn-architectures-explained-716fb197b243

[34] https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/

[35] https://medium.com/@draj0718/convolutional-neural-networks-cnn-architectures-explained-716fb197b243

[36] https://media.geeksforgeeks.org/wpcontent/uploads/20190721025744/Screenshot-2019-07-21-at-2.57.13-AM.png

[37] https://media.geeksforgeeks.org/wpcontent/uploads/20190721025744/Screenshot-2019-07-21-at-2.57.13-AM.png

[38] https://www.tutorialspoint.com/dip/grayscale_to_rgb_conversion.htm

[39] https://medium.com/@danqing/a-practical-guide-to-relub 83ca804f1f7

[40] https://in.mathworks.com/discovery/what-is-matlab.html

[41] https://www.kaggle.com/datasets/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign

[42] https://www.godigit.com/traffic-rules/traffic-signs-in-india

Appendix

- **Data Preprocessing (MATLAB Code)**

```
clc;
folder_path = 'F:/Dataset/Test';
file_list = dir(fullfile(folder_path, '*.png'));
output_folder = 'F:/New/Test1';

for i = 1:length(file_list)
    file_name = file_list(i).name;
    file_path = fullfile(folder_path, file_name);
    image_data = imread(file_path);
    resized_image = imresize(image_data, [32 32]);
    gray_image=rgb2gray(resized_image);
    output_file_name = fullfile(output_folder, file_name);
    imwrite(gray_image, output_file_name);
end
```

- **Proposed Model (MATLAB Code)**

```
clc;
clear all;
% Load the dataset
datasetPath = 'New/Train1';
imds = imageDatastore(datasetPath, 'IncludeSubfolders', true,
'LabelSource', 'foldernames');
```

```
% Split the dataset into training, validation, and testing sets
[imdsTrain, imdsVal, imdsTest] = splitEachLabel(imds, 0.7, 0.15, 0.15,
'randomized');

% Define the CNN architecture
layers = [    imageInputLayer([32 32 1])
    convolution2dLayer(3, 32, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2, 'Stride', 2)
    convolution2dLayer(3, 64, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2, 'Stride', 2)
    convolution2dLayer(3, 128, 'Padding', 'same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2, 'Stride', 2)
    fullyConnectedLayer(256)
    dropoutLayer(0.5)
    fullyConnectedLayer(43)
    softmaxLayer
    classificationLayer];

% Define the training options
options = trainingOptions('adam', ...
    'MiniBatchSize', 128, ...
    'MaxEpochs', 4, ...
```

```
    'ValidationData', imdsVal, ...
    'ValidationFrequency', 10, ...
    'Plots', 'training-progress');

% Train the CNN
net = trainNetwork(imdsTrain, layers, options);
% Test the CNN on the testing set 1
predictedLabels = classify(net, imdsTest);
accuracy = mean(predictedLabels == imdsTest.Labels) * 100;
disp(['Accuracy: ' num2str(accuracy) '%']);

%%
% Calculate the confusion matrix
C = confusionmat(imdsTest.Labels, predictedLabels);
% Create a confusion chart
confusionchart(imdsTest.Labels, unique(trueLabels));
title(' Confusion Matrix ' ) ;

% Visualize the confusion matrix as a heatmap
figure;
customColormap = hot(256) / 255;
heatmap(C, 'Colormap', customColormap, 'ColorbarVisible', 'off',
'XLabel', 'Predicted Labels', 'YLabel', 'True Labels');
title('Confusion Matrix');

%%
% Show Test Images set 1 and their labels
numImages = 40;
```

```
idx = randperm(numel(imdsTest.Files),numImages);

figure
for i = 1:numImages
    subplot(5,8,i)
    I = readimage(imdsTest,idx(i));
    label = predictedLabels(idx(i));
    imshow(I)
    title(char(label))
end
%%
% Accuracy of Test Images set 2
datasetPath2 = 'New/Test1';
imds2 = imageDatastore(datasetPath2);
predictedLabels2 = classify(net, imds2);

%%
% Show Test Images set 2 and their labels
numImages = 40;
idx2 = randperm(numel(imds2.Files),numImages);
figure
for i = 1:numImages
    subplot(5,8,i)
    I = readimage(imds2,idx2(i));
    label = predictedLabels2(idx2(i));
    imshow(I)
    title(char(label))
end
```

Query test (MATLAB Code)

```
clc;
qpath = imgetfile();
inputImage = imread(qpath);
resizedImage = imresize(inputImage, [32 32]);
grayImage=rgb2gray(resizedImage);
predictedLabel = classify(net, grayImage);
disp(['Predicted label: ' char(predictedLabel)]);
imshow(inputImage)
title(char(predictedLabel));
```

Feature Plot (MATLAB Code)

```
clc;
layer=2; %layer=6 for second convolution layer
act = activations(net,imds,layer);
selectedActivationMap = act(:,:,2);
normalizedActivationMap = mat2gray(selectedActivationMap);
figure;
imshow(normalizedActivationMap,'InitialMagnification', 'fit');
title('Feature Plot');
% Plot individual feature maps
numFeatures = size(act, 3);
figure;
for i = 1:numFeatures
    subplot(ceil(sqrt(numFeatures)), ceil(sqrt(numFeatures)), i);
    imshow(act(:, :, i), []);
    title(['Feature Map ', num2str(i)]);
end
```