Traffic Sign Classification Using Deep Learning Based Convolutional Neural Networks

Ahmed J. Abougarair[1], Nagmden Miled Nasser[2], Abdulhamid A. Oun[3]
Electrical and Electronics Engineering, University of Tripoli[1,3]
, Faculty of Engineering, Azzaytuna University[2]

الملخص:

تحتوي إشارات المرور على معلومات مهمة عن حركة المرور على الطرق وتعتبر حجر الزاوية لأنظمة المرور لأنها تضمن سلامة الطريق لكل من المشاة والسائقين. يعتبر تصنيف إشارات المرور جزءًا مهمًا في تشغيل أنظمة مساعدة السائق المتقدمة والمركبات المستقلة وأنظمة النقل الذكية. يساعد تصنيف إشارات المرور على تقليل عدد حوادث الطرق والقرارات الخاطئة التي يتخذها السائقون ويساعد على تعزيز مصداقية المركبات ذاتية القيادة. في هذا الورقة ، تم تصميم نظام لغرض تصنيف إشارات المرور باستخدام الكمبيوتر و بالتحديد تم استخدم كلا من النظام التعلم العميق (Deep Learning) والشبكات العصبية (Convolution Neural Networks). اعتمد النموذج المقترح على نسخة من نموذج في VGG16 و Adam optimization وتم استخدام Google Colaboratory أثناء التدريب لأنه يوفر استخدام الأجهزة الافتراضية مع وحدات معالجة الرسومات التي تساعد في تسريع عملية التدريب بشكل كبير. تظهر نتائج المحاكاة أن النموذج المقترح تم تنفيذه بطريقة ممتازة، حيث وصل إلى قيم قياس تقييم تزيد عن 97٪ مما يؤكد مدى فعالية استخدام تقنية التعليم العميق والشبكات العصبية الصناعية لتصنيف الاشارت المختلفة واتخاذ القرارت الصحيحة.

Abstract:

Traffic sign classification is a challenging computer vision task of high industrial relevance. Traffic signs contain important road traffic information and are considered the cornerstone of traffic systems as they ensure road safety for both pedestrians and drivers. Traffic sign classification is considered a crucial part in the operation of advanced driver assistance systems, autonomous vehicles, and intelligent transportation systems. Traffic sign classification helps reduce the number of road accidents and wrong decisions made by drivers and helps foster the credibility of autonomous vehicles. In this paper, a computer-aided recognition system was designed for the purpose of traffic sign classification.

The system used deep learning, specifically, convolutional neural networks. The proposed model was based on a reduced version of the VGG16 model while integrating batch normalization, dropout and Adam optimization, Google Colaboratory was used during training as it offers the use of virtual machines with GPUs that help speed up the training process immensely. Simulation results show that the proposed model performed in an excellent manner, reaching evaluation metric values of above 97%.

Keywords: DNN, Deep Learning, ITS, Machine Learning.

Introduction

Traffic sign classification is considered a crucial part in the operation of advanced driver assistance systems (ADAS), autonomous vehicles, and intelligent transportation systems (ITS). Traffic signs contain important road traffic information and are considered the cornerstone of traffic systems as they ensure road safety for both pedestrians and drivers, Wirth the great advancements made in convolutional neural networks (CNN) in the field of computer vision, some systems have surpassed human performance and have shown great success in the classification of traffic signs (Smit M. and Chirag P., 2019). This is very useful in ADAS as it helps warn drivers about upcoming traffic signs, which, in turn, helps reduce the number of road accidents and wrong decisions made by drivers. ADAS also help with driving in bad weather conditions and help to overcome drivers misreading traffic signs, which are two major contributing factors to road accidents, With the boom in the automotive industry, the focus has been on the implementation of autonomous vehicles. It is imperative for autonomous vehicles to follow traffic signs and obey traffic rules to ensure road safety and maintain the integrity of the traffic system. Traffic sign classification is important to foster the credibility of autonomous vehicles, otherwise, this may jeopardize the traffic system (Dietmar P.F. Moller and Roland E. Haas,2019).

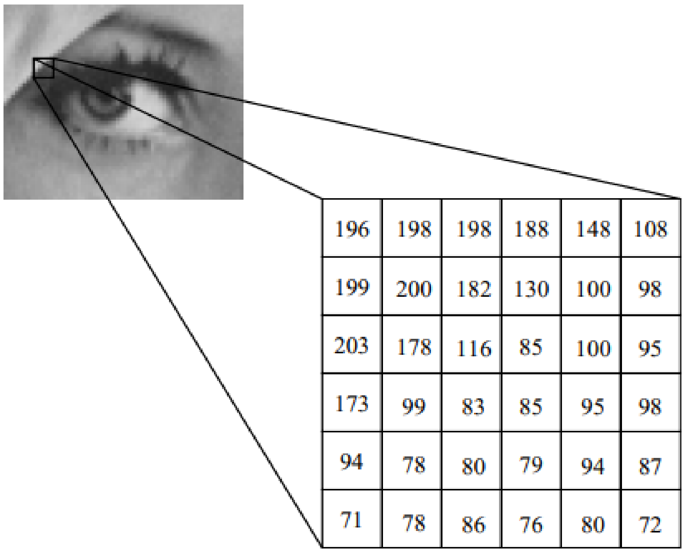
ITS are real-time, safe and accurate integrated traffic management systems. Traffic sign classification systems play a vital role in improving the traffic environment. Timely and accurate transmission of traffic sign information to drivers can help them have sufficient response time to deal with various situations, reduce or avoid traffic accidents and ensure the safety of the people and property of traffic participants (Liu S.,2019). Traffic sign classification will play an important role in intelligent transportation in the future.

Computer Vision

Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images and videos and take actions or make recommendations based on that information. Computer vision, as a scientific discipline, is heavily connected to image processing and directly deals with mathematics and physics and is used in many industries ranging from energy and utilities to manufacturing and automotive. Computer vision trains machines to perform functions similar to that of the human eye; functions like how to tell objects apart, how far away they are, whether they are moving and whether there is something wrong with an image. Due to the rapid development in this field, computer vision systems can surpass human capabilities, as these systems can analyze thousands of images per minute, Computer vision systems need to be fed a great amount of data. The system would then analyze the data repeatedly until it discerns distinctions and ultimately recognizes images. Two essential technologies are used to accomplish this: a machine learning technique called deep learning and convolutional neural networks (CNN)(M. A. Wani,2019).

Digital Images

An image can be defined as a two dimensional continuous function f(x,y), where x and y are considered spatial coordinates. The amplitude of f(x,y) at any coordinate is called the intensity or grey level of the image at that point, To illustrate this, we can see the grey level values of a section of an image in figure1.



196	198	198	188	148	108
199	200	182	130	100	98
203	178	116	85	100	95
173	99	83	85	95	98
94	78	80	79	94	87
71	78	86	76	80	72

Figure 1: Digital Image Representation

The RGB Colour Model

When storing digital images, a colour representation scheme is required. The most common scheme is the Red-Green-Blue (RGB) colour model is often described as a cube with three orthogonal axes, as shown in figure 2.

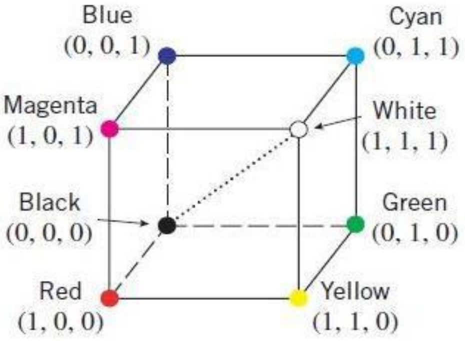


Figure 2: The RGB Model Cube**.**

For image storage, 8-bits are often used for each colour component (24-bits per pixel in total). The standard RGB cube can be quantized and scaled by a factor of 255. With $(2^8)^3$ = 16,777,216 colours. Figure 3 illustrates how the colour values of an image are stored.

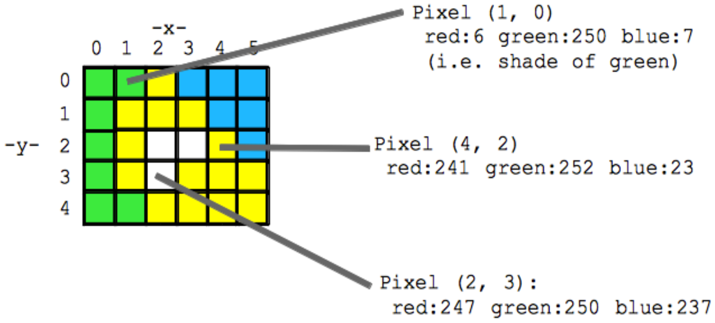


Figure 3: Storage of Colour Values.

Image Classification

Image classification is a subdomain of computer vision in which an algorithm is used to predict the label of an image from a predefined set of labels. Image classification can be considered the fundamental task of computer vision, as it forms the basis for the majority of other computer vision tasks (Kari P. and Anatoly B.,2012).

Image Processing Operations

The following are some image processing operations commonly used in computer vision systems:

Converting to Grayscale

The pixel values of grayscale digital images only carry intensity information. Images of this sort are composed exclusively of shades of gray, varying from black to white. To convert RGB values to grayscale, the following equation is used:

$$Grayscale{=}0.2989\ x\ R{+}0.587\ x\ G{+}0.114\ x\ B \qquad (1)$$

Where R, G, B are the intensity values of the red, green and blue colours respectively. Figure 4 illustrates the grayscale conversion operation.

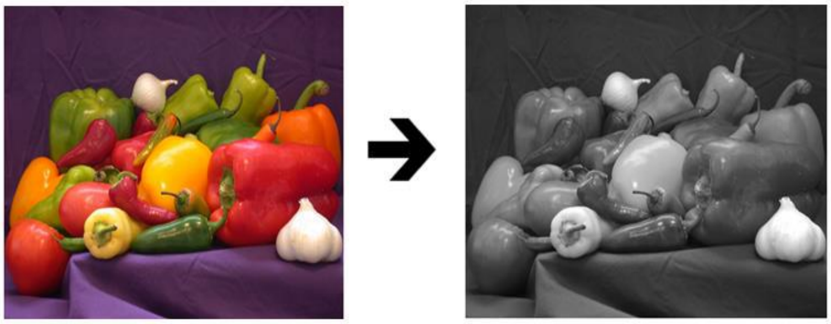


Figure 4: Colour Image Converted to Grayscale**.**

Thresholding

Binarization is the method of converting grayscale images into binary images, A thresholded image g(x, y) is defined as follows:

$$g(x,y) = \begin{cases} 1 \quad if \quad f(x,y) > T \\ 0\ if \quad f(x,y) \leq T \end{cases} \qquad (2)$$

There are two types of thresholding: single and multiple threshold, as illustrated in figure 5.

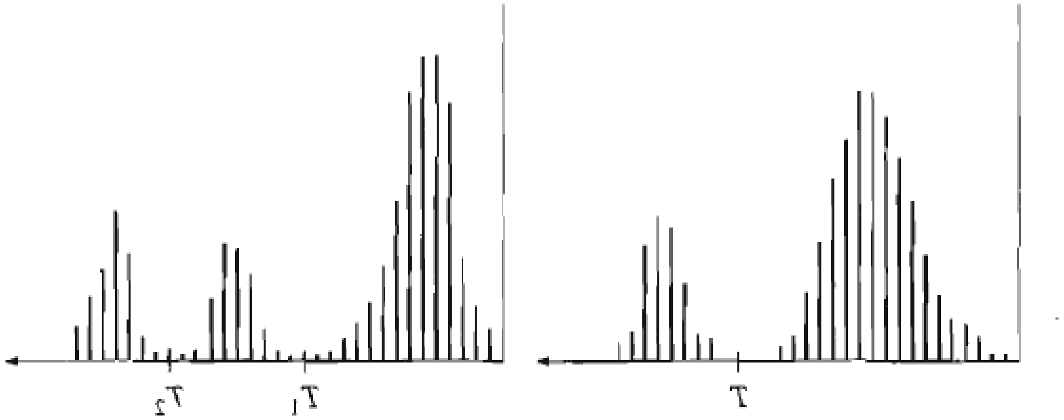


Figure 5: Single and Multiple Thresholds

Edge Detection

Edge detection is an image processing technique used to identify points in a digital image with discontinuities, Figure 6 illustrates an example of edge detection.

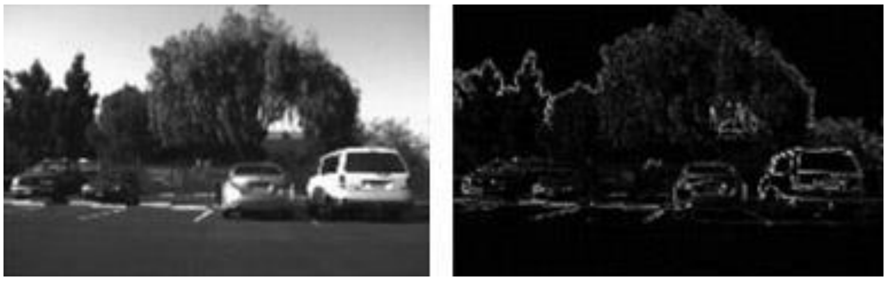


Figure 6: Example of Edge Detection**.**

Bilinear Interpolation

Image interpolation is required when rescaling images. Bilinear interpolation tries to achieve the best approximation of a pixel’s intensity based on the values of surrounding pixels. Figure 7 illustrates how bilinear interpolation on images work.

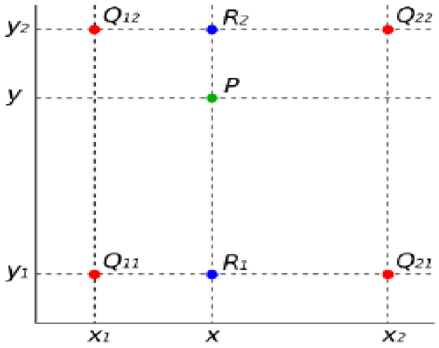


Figure 7: Bilinear Interpolation.

Suppose that we want to find the value of the pixel at point P(x, y). Assuming we know the values of the four points Q11 = (x1, y1), Q12 = (x1, y2), Q21 = (x2, y1), and Q22 = (x2, y2).

Firstly, linear interpolation is done in terms of x This yields:

$$f(x, y_1) = \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x_{\square} - x_1}{x_2 - x_1} f(Q_{21}) \tag{3}$$

$$f(x, y_2) = \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x_{\square} - x_1}{x_2 - x_1} f(Q_{22}) \tag{4}$$

Secondly, linear interpolating is done in terms of y to obtain the desired estimate:

$$f(x,y_1) = \frac{y_2 - y}{y_2 - y_1} f(x,y_1) + \frac{y - y_1}{y_2 - y_1} f(x,y_2) \tag{5}$$

Traffic Sign Classification

Traffic signe classification is the process of automatically recognizing traffic signs. This application of computer vision is used in various systems, Traffic signs can be grouped into five categories as follows: Prohibitory Signs,Danger Signs, Mandatory Signs , Derestriction Signs , Unique Signs (Kari P. and Anatoly B.,2012),The Prohibitory category, shown in figure 8, consists of signs that inform drivers of restrictions:

- .

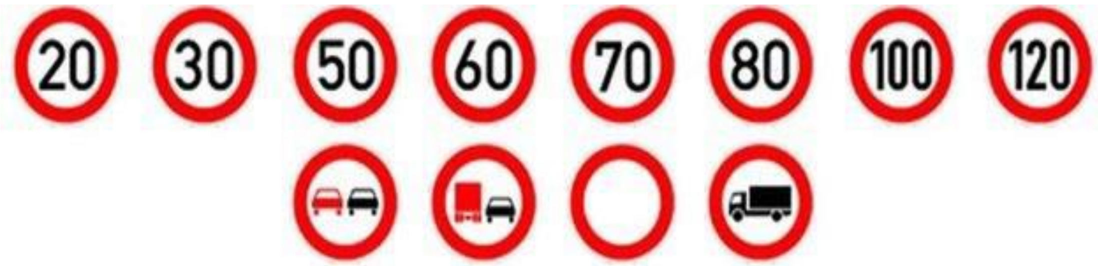


Figure 8: Prohibitory Signs.

The Mandatory category, shown in figure 9, consists of signs that inform drivers of upcoming traffic obligations:

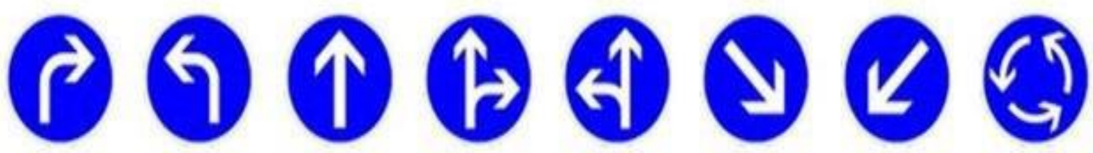


Figure 9: Mandatory Signs.

The Derestriction category, shown in figure 10, consists of signs that signal the end of a specific restriction :

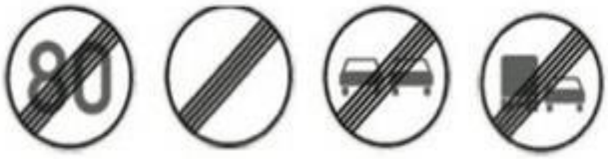


Figure 10: Derestriction Signs.

The Unique category, shown in figure 11, consists of signs that do not fall in any of the previous categories:

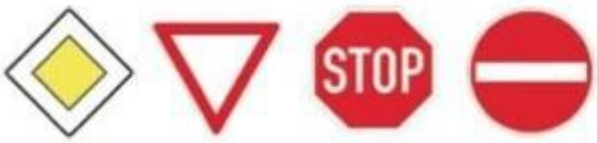


Figure 11: Unique Signs.

Deep Learning

Deep learning (DL), also known as deep neural networks (DNN), is a machine learning technique based on artificial neural networks (ANN) in which multiple layers of processes are used to extract progressively higher level features from data. Deep learning is an important element of data science, which includes statistics and prédictive modeling, (Rafael C. Gonzales and Richard E. Woods,2008), Image classification with machine Learning levurages the potential of algorithmes to learn hidden knowledge from a dataset and with the advent of deep learning, in combination with robust AI hardware and GPUs, outstanding performance can be achieved on image classification tasks. Hence, deep learning brought great successes in the entire field of computer vision, Deep learning algorithms automatically scan the data for features that correlate and combine them to enable faster learning without being explicitly programmed ,DNNs scale with data, meaning, they continue to improve as the size of the data increases. Whereas machine learning and shallow learning (networks with less than 3 hidden layers) tend to converge (M. A. Wani,2019).

Machine Learning

Machine learning (ML) is a data analytics technique that teaches computers to learn from experience. Machine learning algorithms use computational methods to learn information directly from data without being explicitly programmed or without relying on a predetermined equation as a model. The algorithms adaptively improve their performance as the number of samples available for learning increase (I. Goodfellow and Y.Bengio,2016). Machine learning can be defined as a computer program that is said to learn from experience E with respect to some class of tasks T and performance measure P, while the performance at tasks T, as measured by P, improves with experience E (Tom M. Mitchell,1997).

The Task, T

Tasks that are too complex and difficult to be solved using programs written by humans can be tackled using machine learning. In this sense, the task is not the process of learning, it is merely the job that is needed to be undertaken by the program. Learning can be described as attaining the ability to perform a task. For example, if the task is for a robot to drive a car, one could write a program to teach the robot how to drive, or directly program it to perform the driving manually (I. Goodfellow and Y.Bengio ,2016).

The Performance, P

In order to evaluate the capabilities of the machine learning algorithm, a quantitative measure must be designed. This performance is typically dependent on the task that needs to be performed. The function that represents this performance is called a loss or cost function, where the objective is for it to be minimized. Of particular interest is the performance of the machine learning algorithm with data that has not seen before, since this can give an insight on how well the algorithm will perform in the real world. Typically, a separate test set of data is set aside in order to be able to evaluate this performance.

The Experience, E

The data used to train a machine learning algorithm can be considered as its experience. A datapoint is one group of features used that have been quantitatively measured from an object or event. A set of these datapoints is called a dataset. Typically, a machine learning algorithm is able to experience this entire dataset. Broadly speaking, machine learning algorithms can be categorized into two groups: supervised and unsupervised learning.

Supervised Learning

Supervised learning experiences a dataset associated with a label or target. As such, supervised learning observes a dataset for x with an associated value or vector y, thus the algorithm will attempt to predict y from x (Tom M. Mitchell,1997). This paper will mainly focus on supervised learning.

Unsupervised Learning

Unsupervised learning experiences a dataset with varying features, in order to learn useful properties from the dataset. In other words, unsupervised learning attempts to learn explicitly or implicitly a probability distribution p(x), from the random vector x, for which a dataset has been observed (Tom M. Mitchell,1997).

Neural Networks

A neural network is made up of a number of nodes, which are called artificial neurons, as shown in figure 12.

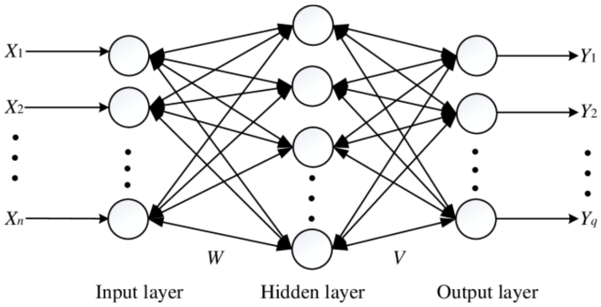


Figure 12: Structure of a Neural Network.

The Artificial Neuron

The artificial neuron can be seen as the fundamental computational block of an ANN. Based on a perceptron, which is a linear binary classifier, the artificial neuron was developed (Martin T. Hagan and Howard B. Demuth). It has a set of N input features, xn, each with a corresponding weight, wn, such that n ∈ {1. . .N} as shown in figure 13.

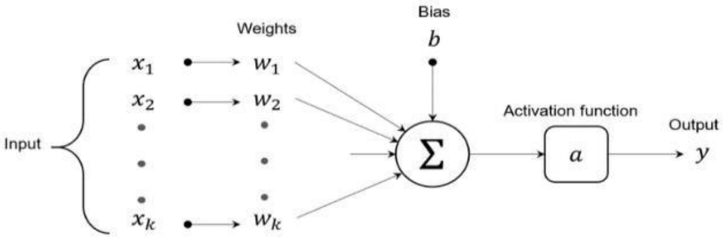


Figure 13: Structure of an Artificial Neural Neuron.

Training

Training is the process of iteratively updating the weights and biases of the neurons depending on the importance of the information provided by the corresponding neuron. Therefore, the training process effectively is attempting to amplify only the most important information, while minimizing the insignificant portions. These correlations are learned blindly. The ANN makes an educated guess and evaluates the performance(I. Goodfellow and Y.Bengio ,2016).

Backpropagation

Backpropagation aims to iteratively adjust the weights in order to reduce the error. Backpropagation computes the output of the ANN. If it matches the specified label then the weights need not be changed. However, if they do not match, the weights contributing to the error need to be changed. Using partial derivatives, backpropagation distributes the blame between the contributing weights and makes the adjustments accordingly Karen S. and Andrew Z., 2015).

Convolutional Neural Network

A convolutional neural network (CNN) is a deep learning algorithm that is most commonly applied to analyze visual imagery. The name convolutional neural network stems from the network's employment of a mathematical operation called convolution. CNNs use convolution in place of general matrix multiplication in at least one of their layers (I. Goodfellow and Y.Bengio ,2016), Convolutional neural networks are composed of two stages: Feature extraction and classification, as shown in figure 14.

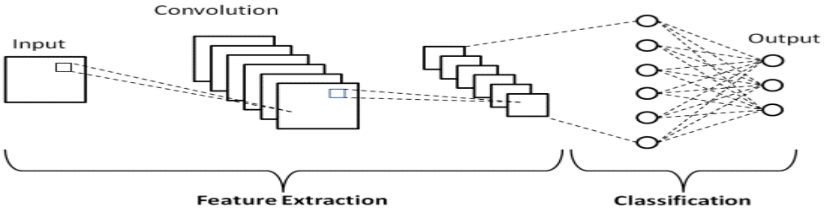


Figure 14: Example of a Convolutional Neural Network**.**

Feature Extraction

The feature extraction stage of a convolutional neural network consists of a sequence of distinct layers that transform the input volume into an output volume through differentiable functions ,The feature extraction stage uses the following distinct layers: Convolutional Layer , Activation Function Layer , Pooling Layer. These layers are stacked together to make up the feature extraction stage, Convolutional and activation function layers are usually attached together followed by an optional pooling layer. In addition to these main layers mentioned above, CNNs may include optional layers like batch normalization usedto improve the training time (L. Luo and Y. Xiong, 2019).

Classification

The classification stage of a CNN is similar to the architecture of a traditional ANN. Neurons are fully connected between different layers. This stage is composed of one or more fully connected layer followed by a classifier ,The task of this stage is to use the detected features in the spatial domain from the previous stage to map each input to a specific class based on the probability distribution among the classes, The distinct layers used in this stage are: Dense Layer ,Dropout (Reddi, M. Zaheer, D. Sachan, S. Kale, and S. Kumar,2018) .

System Implementation & Evaluation

The main steps of the paper are illustrated in the following flowchart in **figure 15**

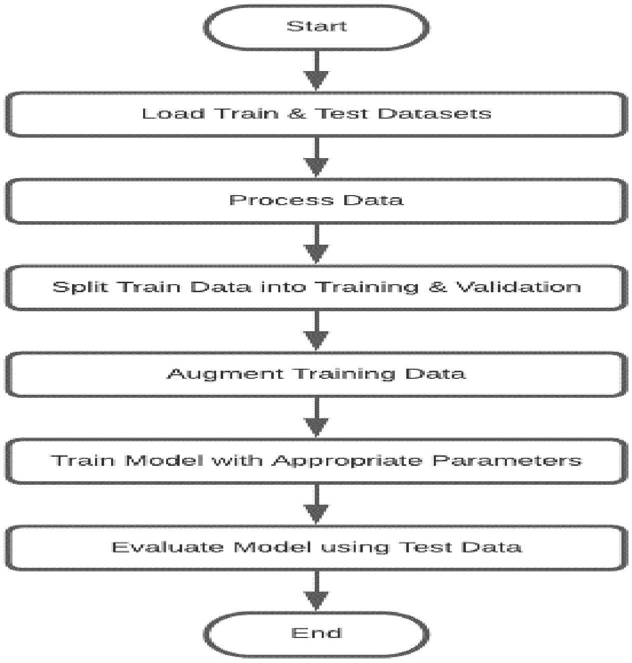

Start
Load Train & Test Datasets
Process Data
Split Train Data into Training & Validation
Augment Training Data
Train Model with Appropriate Parameters
Evaluate Model using Test Data
End

Figure 15: Flowchart

Proposed Model

The proposed system architecture, illustrated in figure 16, is a reduced version of the VGG16 model. The VGG16 model has 13 convolution layers and 3 dense layers. The proposed model has only 4 convolution layers and 2 dense layer. The VGG16 model was designed for object detection with 1000 classes (Karen S. and Andrew Z.,2015). The reduction is done to decrease computational time during training and to overcome any overfitting caused by the complexity of the model. Also thanks to the very large dataset, a smaller model can be used without compromising the performance. Batch normalization and dropout layers were also added. Batch normalization helps stabilize the process and improves speed during training, while dropout helps prevent overfitting, (L. Luo and Y. Xiong,2019).

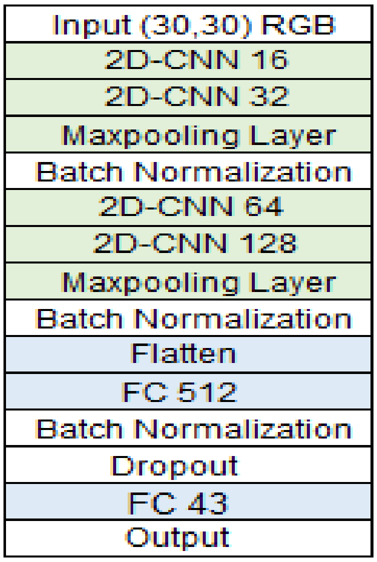


Figure 16: Proposed Model

GTSRB Dataset

The German Traffic Sign Recognition Benchmark (GTSRB) is a classification dataset, The benchmark dataset has the following properties:

- Single-image, multi-class classification dataset.
- 43 Classes.
- 51,839 images in total.
- 39,209 images in the train set.
- 12,630 images in the test set.

Figure 17 illustrates some sample images from the dataset.

Figure 17 : Sample Images from Dataset

Data Preprocessing

The data underwent a few preprocessing operations prior to being used to train and test the model.

- **Imaged rescaling**

  The images were rescaled to 30x30 pixels to match the size of the input layer.

- **Numpy Arrays**

The images and their respective labels were placed into four numpy arrays: train_data, train_labels, test_data, test_labels.

- **Array Shuffling**

train_data and train_labels were shuffled to ensure that all classes are accurately represented when split into training and validation.

- **Splitting Train Data**

Train data was split into 27446 (70%) & 11763 (30%) images for training and validation respectively; which is the recommended ratio.

- **Normalization**

Because neural networks use small weight values, training can be disrupted or slowed down by large integer values. Therefore, the range of pixel intensity values for both the training data and validation data were normalized (range changed to 0-1).

Data Augmentation

The numpy array for the training dataset is augmented to increase the number of training images and to prevent the model from overfitting and memorizing the exact details of the images or memorizing noise.

Table 1: Augmentation Details

Rotation Range	10
Zoom Range	0.15
Width Shift Range	0.1
Height Shift Range	0.1
Shear Range	0.15

Training Parameters

Parameters for the 4 convolution layers:

Table 2: Convolution Layer Parameters

Kernal Size	All 3X3
Number of Filters	16,32.64,128
Stride	All 1X1
Padding	All Valid
Activation Function	All ReLU

Parameters for the 2 Maxpooling layers:

Table 3: Maxpooling Layer Parameters

Pool Size	Both 2x2

Parameters for the 2 Dense Layers:

Table 4: Dense Layer Parameters

Number of Nodes	512, 43
Activation Function	ReLU, Softmax

Other Training Parameters

Table 5 :Other Training Parameters

Dropout	0.5
Learning Rate	0.01
Epochs	30
Batch Size	32
Loss Function	Cross Entropy
Optimizer	Adam

System Implementation

This Paper was implemented using python programming language version 3.9 with the following libraries: TensorFlow , Keras , Keract OpenCv , Pillow , Numpy , Pandas , Sklearn , Tkinter .Google Colaboratory was used during the training because it offers the use of virtual machines with GPUs that help speed up the training process.

Model Training

Figure 18 presents the learning curves during training,The figure illustrates the training progress showing the accuracy and loss per epoch during both the training and validation.

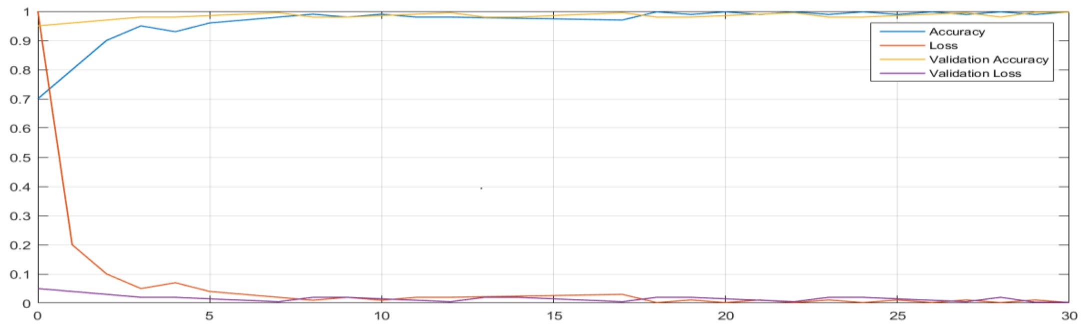
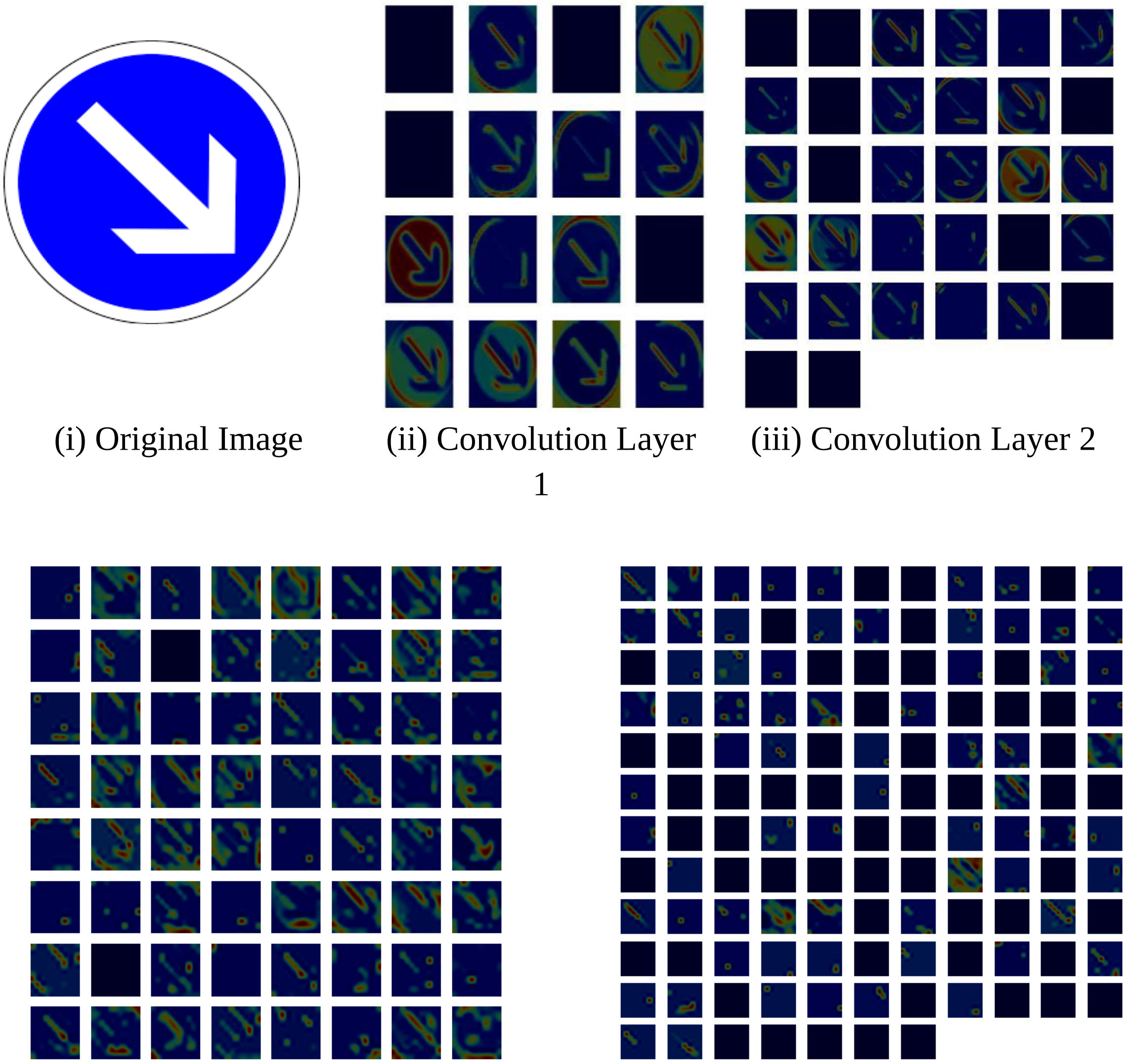


Figure 18: Training and Validation Learning Curves.

We can see that the generalization gap between the training loss and validation loss curves is miniscule and that both loss curves reach a stable point, There is also no gap found between the training accuracy and validation accuracy curves, These indicate that the model was well fit .The training accuracy reached 99.69**%.** While the validation accuracy reached 99.96%.

Visualizing Activations

In this section, the convolution layer outputs are visualized to help understand what happens during the feature extraction. Figure 17 illustrates the outputs of each convolution layer.



(i) Original Image (ii) Convolution Layer 1 (iii) Convolution Layer 2

(iv) Convolution Layer 3 (v) Convolution Layer 4

Figure 19: Convolution Layer Features

We can see that the early stages have smaller receptive fields and extract a small amount of features, while the later stages have larger receptive fields and extract a lot more features.

Performance Evaluation

The model's accuracy when evaluated on the test dataset reached 98.48% A detailed classification report is drawn up showing the evaluation metric values such as, precision, recall and f1-score for each class. The evaluation metrics are:

Precision = TP/(TP + FP)

Recall = TP/(TP + FN).

$$f1 - score = \frac{(2 \times Precision \times Recall)}{(Precision + Recall)}$$

Where TP, TN, FP, FN are:

TP: true positive is an outcome where the model correctly predicts the positive class.

TN: true negative is an outcome where the model correctly predicts the negative class.

FP: false positive is an outcome where the model incorrectly predicts the positive class

FN: false negative is an outcome where the model incorrectly predicts the negative class.

Detailed Classification:

Table 7: Prohibitory Signs Classification Report

Class Name	Number of Images	Precision	Recall	F1-Score
Speed Limit (20km/h)	60	79%	100%	88%
Speed Limit (30km/h)	720	100%	100%	100%
Speed Limit (50km/h)	750	99%	100%	100%
Speed Limit (60km/h)	450	96%	98%	97%
Speed Limit (70km/h)	660	100%	100%	100%
Speed Limit (80km/h)	630	98%	99%	98%
Speed Limit 100km/h)	450	100%	100%	100%
Speed Limit (120km/h)	450	100%	97%	98%

Overtaking Prohibited	480	100%	100%	100%
Overtaking Prohibited for Trucks	660	100%	100%	100%
All Vehicles Prohibited	210	99%	100%	99%
Trucks Prohibited	150	100%	100%	100%

Table 8: Danger Signs Classification Report

Class Name	Number of Images	Precision	Recall	F1-Score
Right of Way at Next Intersection	420	96%	100%	98%
General Caution	390	99%	93%	96%
Dangerous Curve Left	60	97%	100%	98%
Dangerous Curve Right	90	98%	100%	99%
Winding Road	90	83%	100%	91%
Bumpy Road	120	99%	84%	91%
Slippery Road	150	99%	99%	99%
Road Narrows on Right	90	97%	98%	97%
Road Work	480	100%	97%	98%
Traffic Light	180	83%	100%	91%
Pedestrian Crossing	60	86%	50%	63%
Children Crossing	150	99%	99%	99%
Bicycle Crossing	90	85%	100%	92%
Beware of Ice/Snow	150	99%	83%	90%
Wild Animals Crossing	270	100%	100%	100%

Table 9 :Mandatory Signs Classification Report

Class Name	Number of Images	Precision	Recall	F1-Score
Turn Right	210	99%	100%	100%
Turn Left	120	100%	100%	100%
Straight Only	390	99%	98%	99%
Straight or Right Only	120	98%	100%	99%
Straight or Left Only	60	100%	98%	99%
Keep Right	690	100%	100%	100%
Keep Left	90	99%	98%	98%
Roundabout Mandatory	90	96%	98%	97%

Table 10: Derestriction Signs Classification Report

Class Name	Number of Images	Precision	Recall	F1-Score
Main Road	690	100%	98%	99%
Yield	720	100%	100%	100%
Stop	270	100%	100%	100%
No Entry	360	100%	100%	100%

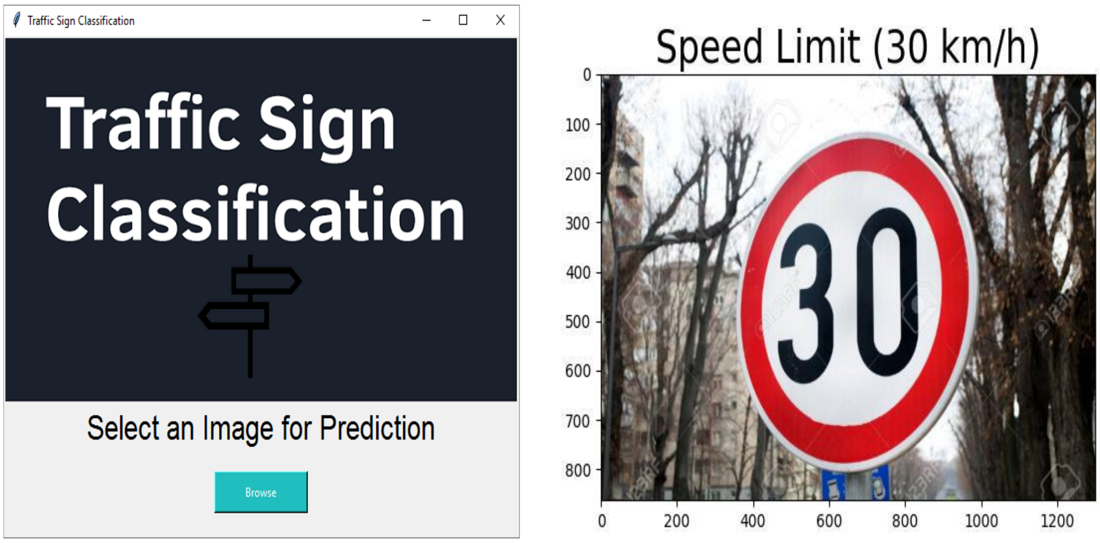
Table 11: Unique Signs Classification Report

Class Name	Number of Images	Precision	Recall	F1-Score
End of Speed Limit (80 km/h)	150	100%	97%	99%
End of All Restrictions	60	100%	100%	100%
End of Overtaking Restriction for All Vehicles	60	100%	100%	100%
End of Overtaking Restriction for Trucks	90	99%	100%	99%

The values, as seen in the classification report, are considered excellent and indicate that the model effectively operated in the presence of unseen data, the precision value of the Speed Limit (20km/h) sign is considerably lower than the other signs in the prohibitory category. This is due to the fact that this sign is considered underrepresented in comparison to the other signs. A low precision and high recall value indicates that the problem lies with false positive predictions, The pedestrian crossing sign has a very low recall value. A low recall value indicates that the problems lie with false negative predictions, as opposed to the 20km/h speed limit sign. The false negative predictions can also be traced back to the fact that this sign is also underrepresented.

GUI Design

A graphical user interface was created to be used in a local environment. Figure 18 shows the GUI interface and output, The GUI simply prompts the user to browse the directory and choose an image to be tested. The program then runs the classifier previously designed and displays the image labelled with the predicted class.


Traffic Sign Classification
Traffic Sign Classification
Select an Image for Prediction
Browse
Speed Limit (30 km/h)
0
100
200
300
400
500
600
700
800
0
200
400
600
800
1000
1200

(i) GUI Main Screen ii) Example of Prediction using GUI

Figure 20: Graphical User Interface

Conclusions

Traffic sign recognition is a challenging computer vision task of high industrial relevance. This paper discussed how to design and implement an effective recognition system for traffic sign classification using convolutional neural networks. The proposed model was trained and its performance evaluated using the GTSRB's train and test datasets respectively.

- Our model achieved an accuracy of 98.48% and a macro precision, recall and f1-score of 97%, despite it being considered small in comparison to other models. This is due to the large dataset used with the aid of the augmented images.
- The unbalanced distribution of the data lead to some classes being underrepresented and ultimately achieved evaluation metric values lower than that of other classes.
- Batch normalization and the dropout layer integrated into the model helped increase training speed and overcome overfitting problems respectively. While Adam optimization was utilized for faster adaptation.

References

[1] Smit M. and Chirag P. (2019). "CNN based Traffic Sign Classification using Adam Optimizer" in the International Conference on Intelligent Computing and Control Systems,30-31

[2] Dietmar P.F. Moller and Roland E. Haas (2019), "Guide to Automotive Connectivity and Cybersecurity", 1st edition.

[3] Liu S. (2019), "A Traffic Sign Image Recognition and Classification Approach Based on Convolutional Neural Network", in the 11th International Conference on Measuring Technology and Mechatronics Automation.

[4] Kari P. and Anatoly B. (2012), "Real-time Computer Vision with Open Cv", 1st edition, The Newton Institute.

[5] M. A. Wani, (2019) "Advances in Deep Learning", 1st edition.

[6] L. Luo and Y. Xiong, (2019), "Adaptive gradient methods with dynamic bound of learning rate," in Proceedings of the 7th International Conference on Learning Representations, New Orleans, Louisiana.

[7] Tom M. Mitchell (1997), "Machine Learning", 1st edition, McGrew-Hill.

[8] I. Goodfellow and Y. Bengio (2016), "Deep Learning", MIT press Cambridge, vol 1.

[9] Rafael C. Gonzales and Richard E. Woods (2008), "Digital Image Processing", 3rd edition, Prentice Hall.

[10] Martin T. Hagan and Howard B. Demuth, "Neural Network Design", 2nd edition, eBook.

[11] Karen S. and Andrew Z. (2015), "Very Deep Convolutional Networks for Large-Scale Image Recognition" in the ICLR Conference.

[12] L. Luo and Y. Xiong, (2019), "Adaptive gradient methods with dynamic bound of learning rate," in Proceedings of the 7th International Conference on Learning Representations, New Orleans, Louisiana.

[13] S. Reddi, M. Zaheer, D. Sachan, S. Kale, and S. Kumar (2018), "Adaptive methods for nonconvex optimization," in Proceeding of 32nd Conference on Neural Information Processing Systems (NIPS 2018).

[14] Karen S. and Andrew Z. (2015), "Very Deep Convolutional Networks for Large-Scale Image Recognition" in the ICLR Conference.