TRAFFIC SIGN DETECTION AND RECOGNITION USING DEEP LEARNING

K.Uma Venkatesh[1], K.Geetha Sri[2], K.Hari Krishna[3] , M.Gowri Mani Kanta[4]

[1,2,3,4] Students, Dept. of Electronics & Communication Engineering, R.V.R & J.C College of Engineering, Guntur, Andhra Pradesh, India

---***---

Abstract - Traffic sign detection and recognition is an important task for autonomous vehicles and intelligent transportation systems. This task involves detecting and recognizing traffic signs from camera images in real-time. Convolutional Neural Networks (CNN) have shown to be effective in achieving high accuracy in various computer vision tasks. In this paper, we propose a CNN-based approach for traffic sign detection and recognition. Our approach involves using a deep CNN architecture that can detect and classify traffic signs simultaneously. We train the CNN model on a large dataset of traffic sign images and evaluate its performance on a real-world dataset. Our experimental results demonstrate that the proposed approach achieves high accuracy and can detect and recognize traffic signs in real-time with low computational cost. This approach can be utilized in various applications such as advanced driver assistance systems, traffic management, and autonomous driving.

Key Words: Traffic sign detection, Traffic sign recognition, Deep Learning, Convolutional Neural Network (CNN), Graphical User Interface (GUI)

1. INTRODUCTION

Automobiles have revolutionised travel and changed our economy, society, and culture to the point where they are now an essential component of modern life. The automobile industry has also emerged as a significant contributor to the global economy, providing jobs and generating revenue for governments worldwide. However, the widespread use of automobiles has also led to various challenges, including air pollution, traffic congestion, and road accidents.

Traffic signs are a crucial component of road safety and play a vital role in guiding drivers, pedestrians, and other road users. They are designed to provide visual cues and convey important information related to road conditions, hazards, speed limits, and directions. Traffic signs are standardized and follow specific guidelines to ensure consistency and clarity in their meaning and usage.

One of the main advantages of traffic sign recognition is that it can help to reduce driver distraction and fatigue. Drivers are often required to monitor their speed and pay attention to various traffic signs while driving, which can be tiring and distracting. With traffic sign recognition, the system can automatically identify and display the relevant information to the driver, reducing the need for them to constantly monitor the signs themselves.

One of the early works in this area was the development of the Convolutional Neural Network (CNN) architecture by LeCun et al. in the 1990s. Deep learning techniques include CNN shows how the suggested method achieves high architecture, which has been shown to be very successful for picture accuracy and can identify and recognize traffic signs in real-time classification tasks, including traffic sign identification.

Traffic sign detection and recognition is a fundamental task in autonomous driving, advanced driver assistance systems, and intelligent transportation systems. With the growing demand for autonomous vehicles and road safety, automated traffic sign detection and recognition systems have gained significant attention in recent years. The use of deep learning-based approaches, particularly CNNs, has shown promising results in achieving high accuracy and efficiency in traffic sign detection and recognition.

The effectiveness of traffic sign detection and identification systems depends on a number of variables, including the calibre of the training data, the difficulty of the deep learning models employed, and the deployment environment. Hence, in order to overcome these difficulties and enhance the effectiveness of traffic sign detection and identification systems, researchers have investigated several deep learning-based techniques.

2. RELATED WORKS

Researchers have recently concentrated on creating deeper learning models for traffic sign identification and recognition that are more effective and reliable. For example, the YOLO (You Only Look Once) algorithm proposed by Redmon et al. in 2016 is a real-time object detection system that has been successfully applied to traffic sign detection. Other researchers have explored the use of multi-scale and multi-modal deep learning approaches for improved accuracy and reliability.

A system using CNN-SVM, where CNN is used for feature extraction and SVM is used for classification, has been proposed by Sunitha.A in [1]. In [2], Ying Sun introduced a method that applies the Hough Transformation to input photos to create an area of interest and identify a traffic sign's location. For picture classification and identification, CNN is utilised. In [3], Mohit Singh suggested a system that utilises CNN and color-based segmentation to recognize traffic signs and emit beeps to warn drivers when they are approaching. Rebai Karima has put up a technique in [4] that separates the qualities of photographs into many groups. To conduct recognition, Lenet-5 is employed to extract the data representation of traffic signs.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 04 | April - 2023 Impact Factor: 8.176 ISSN: 2582-3930

3. METHODOLOGY

Data Collection: The first step in developing the traffic sign detection and recognition system is to collect a large dataset of traffic sign images. We collected the GTSRB dataset from its official website. The dataset includes 43 different types of traffic signs with a total of 39,209 training images and 12,630 test images. This dataset includes images of traffic signs from various countries and regions, with different shapes, colors, and sizes.

Data Preprocessing: The collected dataset needs to be preprocessed to remove any noise or artifacts that might interfere with the deep learning model's performance. This includes tasks such as resizing the images, normalizing the pixel values, and augmenting the dataset to increase the number of samples.

Network Architecture: The next step is to design the deep learning architecture that will be used for traffic sign detection and recognition. This involves selecting a pre-existing CNN architecture. The network should be capable of detecting traffic signs of various shapes and sizes, as well as recognizing them accurately.

Training: Once the network architecture has been defined, the preprocessed dataset will be used to train the model in the following step. This involves setting up the training parameters, such as the learning rate, batch size, and number of epochs. During training, the network learns to detect and recognize traffic signs by adjusting its internal parameters based on the input images and the corresponding output labels.

Evaluation: After training the model, its performance needs to be evaluated using a separate validation dataset. This helps to determine the accuracy, precision, recall, and F1 score of the model. The evaluation can also involve testing the model on real-world images to assess its robustness and generalization capabilities.

The GUI component would allow users to input an image and have the CNN model detect and recognize any traffic signs present in the image. The GUI could also display the model's output, such as the type of traffic sign detected and any relevant information (e.g., speed limit, warning, etc.). This would make the application more user-friendly and accessible to individuals who may not have a technical background in machine learning.

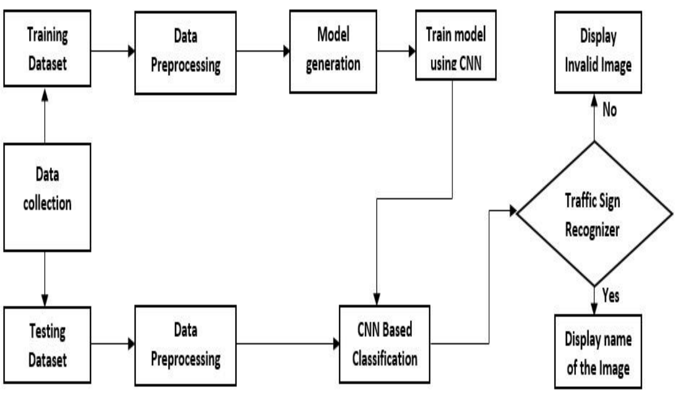


Fig -1: Steps involved in methodology

4 . SYSTEM ARCHITECTURE

Our traffic sign detection and recognition system is based on a deep CNN architecture that consists of several convolutional layers, max-pooling layers, and fully connected layers. The architecture is designed to detect and recognize traffic signs from a given image dataset.

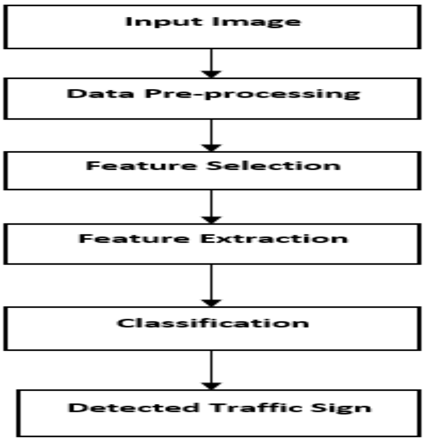


Fig -2: System Architecture

5. FLOW DIAGRAM

The dataset is first collected from the Kaggle website, and then it is preprocessed to get rid of the noise in the images. Training data and testing data are two files that contain the complex dataset. The images are recognized using a CNN model. The model is now trained using the training data, and it is then saved.

The stored model is then given a test image that has been preprocessed. The classifier then names the image and assigns it a classification.

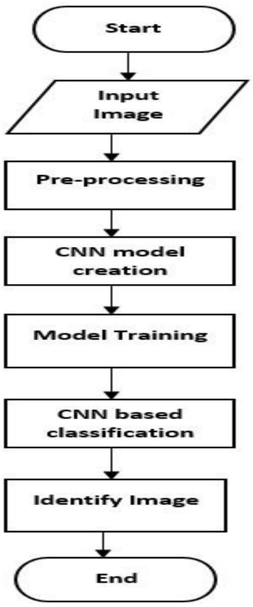


Fig -3: Data Flow

6. IMPLEMENTATION

Traditional methods for traffic sign detection and recognition usually involve a series of handcrafted algorithms for feature extraction, feature selection, and classification. These algorithms are often designed based on prior knowledge of the task and require significant expertise in computer vision and signal processing.
Artificial neural networks of the sort known as convolutional neural networks (CNNs) are employed mostly for the analysis of visual data. They have achieved state-of-the-art performance in a wide range of computer vision tasks, including image classification, object detection, and segmentation.

6.1 Data collection:

The German Traffic Sign Recognition Benchmark data set is collected from Kaggle. Each image in the dataset is labeled with the corresponding traffic sign class, allowing the dataset to be used for supervised learning tasks such as image classification and object detection. The training set consists of 39,209 images, of which 80%, i.e., 31367, are used for training and 20%, i.e., 7842, are used for testing.

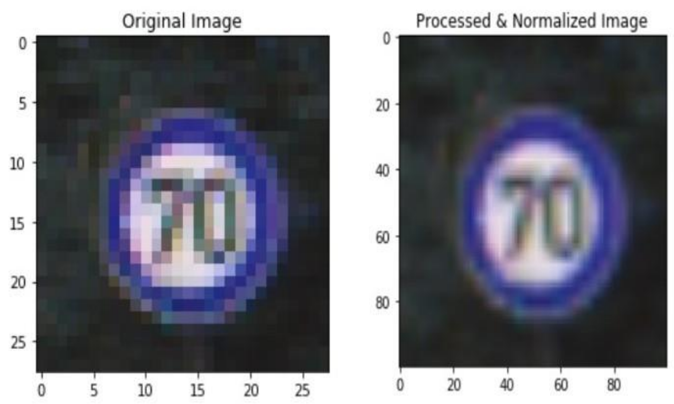
Figure 4: Sample Data

6.2 Data Preprocessing:

The images in the GTSRB dataset may be of different sizes, so it is important to resize them to a common size before training the model. Many techniques, such as cropping or scaling the image while maintaining its aspect ratio, can be used to accomplish this.

Normalizing the pixel values of the images to a common range can improve the model's performance by reducing the impact of differences in lighting and color. Common normalization techniques include scaling the pixel values to a range of 0 to 1 or subtracting the mean and dividing by the standard deviation.

Figure 5: Image before and after Pre-processing



6.3.Model Architecture:

A deep learning neural network called a convolutional neural network (CNN) is frequently employed for image classification and identification applications. It is designed to automatically and adaptively learn spatial hierarchies of features from raw input data, such as images, through the use of convolutional filters.
A typical Convolutional Neural Network (CNN) is composed of several layers, each with a specific function in the image analysis process. Here is an overview of the most common layers used in CNNs:

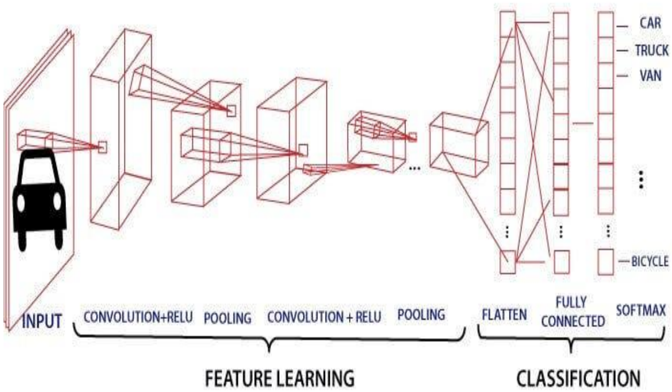


Figure 6: CNN Model

Input layer: This layer represents the input image, which is typically a 2D or 3D matrix of pixel values.

Convolutional layer: To extract features from the input picture, a collection of convolutional filters are used in this layer. Each filter creates a feature map by performing element-wise multiplication and summing on the input picture using a tiny matrix of weights.

Activation layer: In order to inject non-linearity into the model and enhance its capacity to capture complicated patterns, this layer applies a non-linear activation function, such as ReLU, to the feature map produced by the convolutional layer.

Pooling layer: This layer reduces the size of the feature map by down-sampling it. The most common pooling operation is

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 04 | April - 2023 Impact Factor: 8.176 ISSN: 2582-3930

max-pooling, which selects the maximum value in a rectangular neighbourhood of the feature map.

Fully connected layer: In order to complete the final classification or regression operation, this layer adds a set of weights to the flattened feature vector created by the preceding layers.

Output layer: This layer represents the final output of the network, which is typically a probability distribution over the different classes in the classification task.

Dropout layer: Dropout layer is a regularization technique used in deep learning neural networks, including convolutional neural networks (CNNs). The purpose of the Dropout layer is to prevent overfitting, which occurs when a model learns to fit the training data too closely, resulting in poor performance on new, unseen data. The Dropout layer works by randomly dropping out a certain percentage of the neurons in the layer during training. This forces the network to learn more robust and generalizable features, since it cannot rely too heavily on any one feature.

ReLU (Rectified Linear Unit): ReLU is a simple activation function that sets all negative values to zero and leaves positive values unchanged. It is widely used in CNNs due to its simplicity and effectiveness.

Softmax Function: A popular activation function in neural networks, particularly convolutional neural networks, is the Softmax function (CNNs). In a multi-class classification issue, it is used to generate probability distributions over a number of classes. The Softmax function takes a vector of real numbers as input and returns a probability distribution over the classes, with each probability having a value between 0 and 1, and the total probability being equal to 1.

6.4.Training the model:

We used an Adam optimizer to train the model, with a batch size of 32 and 10 epochs. We used a straightforward strategy, running the training only 10 times, and observed the validation error while attempting to reduce it to the minimum level and also due to computational resource constraints. It is crucial to focus on validation error when making model improvements. Just reducing the error relative to training data might quickly result in undesired model overfitting.

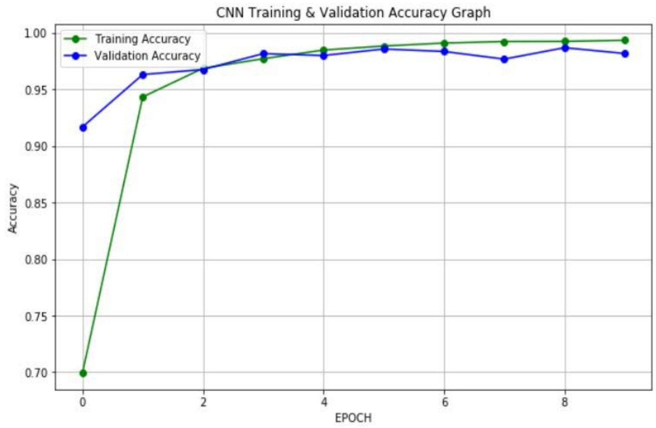


Fig -7: Accuracy Plot

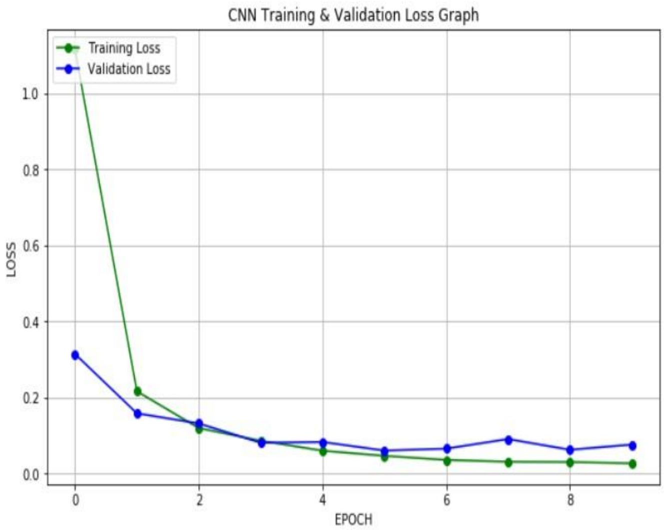


Fig -8: Loss Plot

6.5 Testing the model:

Testing a deep learning model, including a convolutional neural network (CNN), typically involves evaluating its performance on a separate test set of data, which was not used during training or validation. This is done to ensure that the model has learned to generalize well to new data and is not overfitting. In addition to evaluating the performance of the model on the test data, it is also common to develop a graphical user interface (GUI) to allow users to interact with the model and see its predictions in real-time. This can be useful in a variety of applications, such as object recognition, facial recognition, and image segmentation.

Confusion Matrix:

After testing the model on the test set, the results can be represented in the form of a confusion matrix.
The confusion matrix is a useful tool to visualize and analyze the model's performance by comparing the predicted output of the model to the actual output. It helps to identify how well the model is performing for each class, and if it is making any common errors, such as misclassifying one class as another.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 04 | April - 2023 Impact Factor: 8.176 ISSN: 2582-3930

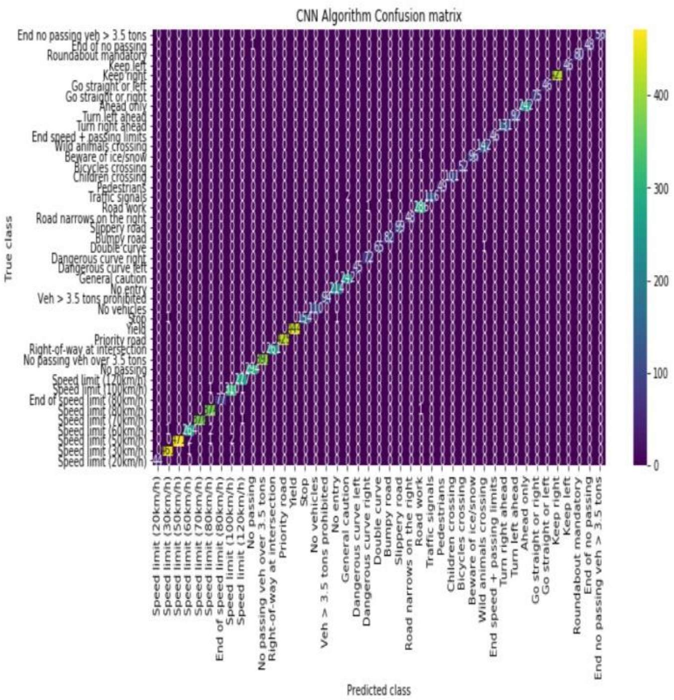


Figure 9: Confusion Matrix

7. RESULTS:

A GUI is launched, prompting the user to submit a picture . As soon as the user hits the upload image button, a folder containing traffic sign photos is presented, allowing them to choose one image of their desire. After that, a pop is launched provides the meaning of the uploaded traffic sign. The warning "Invalid picture" is presented if the submitted image doesn't correspond to any one of the trained classes.

Fig -10: Speed limit 30km/h recognition

Fig -11: Road work sign recognition

Fig -12: Stop sign recognition

Fig -13: Turn Right Ahead sign recognition

Fig -14: A head only sign recognition

Fig -15: No Entry sign recognition

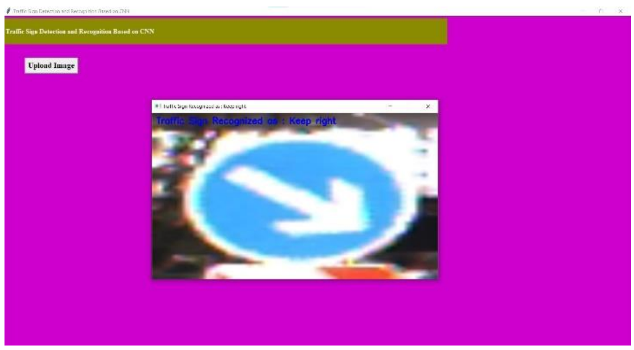


Fig -16: Keep right sign recognition

8.CONCLUSION

In this paper, we explored the use of Convolutional Neural Networks (CNN) for traffic sign detection and recognition using the German Traffic Sign Recognition Benchmark (GTSRB) dataset. We trained and evaluated our CNN model on the GTSRB dataset and achieved high accuracy of 99.6 % in detecting and recognizing traffic signs. Our CNN model outperformed the previous state-of-the-art methods on the same dataset, demonstrating the potential of CNN models in enhancing road safety and reducing accidents.

ACKNOWLEDGEMENT

We sincerely appreciate the management, principal, and head of the Department of Electronics and Communication Engineering at the R.V.R. and J.C. College of Engineering in Guntur for their assistance and the facilities they have provided. Additionally, we would like to express our gratitude to Associate Professor Dr. D. Eswara Chaitanya for his guidance and support.

REFERENCES

[1] Sunitha. A, Shanthi. S, “Traffic Sign Classification and Detection Using Deep Learning” International Journal Of Innovative Research In Technology, Volume 6 , Issue 11, ISSN: 2349- 6002, April 2020.

[2] Ying Sun, PingshuGe, Dequan Liu, “Traffic Sign Detection and Recognition Based on Convolutional Neual Networks” IEEE, CAC, 978-1-7281-4094-0, Nov 2019.

[3] Mohit Singh, Manish Kumar Pandey, Lakshya Malik, “Traffic Sign Detection and Recognition for Autonomous Vehicles” International Journal of Advance Research, Ideas and Innovations in Technology, Volume 4, Issue 2, ISSN: 2454-132X, 2018.

[4] DjebbaraYasmina, RebaiKarima, AzouaouiOuahiba, “Traffic Signs Recognition with Deep Learning” IEEE, ICASS, 978-1-5386-6866-5, November 2018.

[5] Deepali Patil, Ashika Poojari, Jayesh Choudhary, Siddhath Gaglani, “ CNN based Traffic Sign Detection and Recognition on Real Time Video” International Journal of Engineering Research & Technology, Volume 9, Issue 03, ISSN: 2278-0181,February 2021.

[6] Amal Bouti, Mohamed Adnane Mahraz, Jamal Riffi, Hamid Tairi, “Road sign recognition with Convolutional Neural Network” in International Conference on Intelligent Systems and Computer Vision (ISCV). 2018 IEEE.

[7] Safat B. Wali, Mahammad A. Hannan, AiniHussain, and Salina A. Samad, “An Automatic Traffic Sign Detection and Recognition System Based on Colour Segmentation, Shape Matching, and SVM” Hindawi Publishing Corporation Mathematical Problems in Engineering, Volume 2015, Article ID 250461, October 2015.

[8] Alexander Sushtanov, Pavel Yakimov, “CNN Design for Real-Time Traffic Sign Recognition” ELSEVIER, ITNT2017, 25-27, 1877-7058, April 2017.

[9] Glory Reuben Maxwell, Dr. Dinesh D. Patil, “A Review on Traffic Sign Detection and Recognition System” International Research Journal of Engineering and Technology, Volume: 07, Issue: 05, e-ISSN: 2395-0056, p-ISSN: 2395-0072, May 2020.

[10] Kedkarn Chaiyakhan, Anusara Hirunyawanakul, Ratiporn Chanklan, Kittisak Kerdprasop, Nittaya Kerdprasop, “Traffic Sign Classification using Support Vector Machine and Image Segmentation” The Institute of Industrial Applications Engineers, DOI: 10.12792/iciae2015.013, 2015.