

TRAFFIC-SIGN-RECOGNIZATION-USING-CNN-ALGORITHM-CONVENTIONAL-NEURO-NETWORK

D.Prathap reddy

UG Student, Department of Computer Science and Engineering. School of Engineering and Technology.

D.Udaykiran

UG Student, Department of Computer Science and Engineering. School of Engineering and Technology.

CH.Sri sai gnanesh

UG Student, Department of Computer Science and Engineering. School of Engineering and Technology.

Mr.M.Suresh

Assistant Professor, Department of Computer Science and Engineering. School of Engineering and Technology.

Dhanalakshmi Srinivasan University, Trichy-621112, Tamil Nadu, India

Abstract : Traffic sign recognition plays a crucial role in improving road safety and supporting intelligent transportation systems. With the rapid growth of vehicles and increasing traffic complexity, manual recognition of traffic signs by drivers may lead to errors due to factors such as

distraction, poor lighting conditions, weather changes, and high driving speed. To overcome these challenges, automated traffic sign recognition systems using Machine Learning techniques have gained significant importance.

This project focuses on Traffic Sign Recognition using a Convolutional

Neural Network (CNN) algorithm and a Conventional Neural Network (ANN). The proposed system processes traffic sign images, applies image preprocessing techniques such as resizing and normalization, and classifies the signs into different categories. ANN is used as a baseline model for basic classification, while CNN is employed for efficient feature extraction and accurate recognition of traffic signs from images.

The performance of both ANN and CNN models is analyzed and compared based on accuracy and efficiency. Experimental results show that CNN outperforms the conventional neural network in recognizing traffic signs due to its ability to capture spatial and visual features effectively. This system can be integrated into driver assistance systems and autonomous vehicles to enhance road safety, reduce human error, and support smart transportation solutions.

I. INTRODUCTION

Road safety is a major concern worldwide due to the increasing number of vehicles and traffic accidents. Traffic signs provide essential information such as speed limits, warnings, and road regulations that help drivers follow traffic rules and ensure safe driving. However, drivers may fail to notice or correctly interpret traffic signs because of

distractions, fatigue, poor visibility, weather conditions, or high vehicle speed. This limitation creates a need for an automated system that can accurately recognize traffic signs.

Traffic Sign Recognition (TSR) is an important application of Machine Learning and Computer Vision, especially in the development of Advanced Driver Assistance Systems (ADAS) and autonomous vehicles. TSR systems automatically detect and classify traffic signs from road images, helping vehicles respond appropriately in real-time situations.

This project presents a Traffic Sign Recognition system using a Convolutional Neural Network (CNN) algorithm and a Conventional Neural Network (ANN). The system uses image preprocessing techniques to enhance image quality and improve recognition accuracy. ANN is used as a basic model to understand classification performance, while CNN is applied to extract complex visual features from traffic sign images. By comparing both models, this project demonstrates the effectiveness of CNN in achieving higher accuracy and reliability.

The proposed system aims to improve road safety, reduce human error, and support intelligent transportation systems by providing accurate and automated traffic sign recognition.

II. RELATED WORK

Traffic Sign Recognition (TSR) has been an active research area in the fields of Machine Learning, Computer Vision, and Intelligent Transportation Systems. Earlier research mainly focused on traditional image processing techniques combined with classical machine learning algorithms. These methods relied on handcrafted features such as color, shape, and edge detection to identify traffic signs. Techniques like Histogram of Oriented Gradients (HOG), Scale-Invariant Feature Transform (SIFT), and Support Vector Machines (SVM) were commonly used. Although these approaches worked reasonably well under controlled conditions, their performance degraded in real-world environments due to variations in lighting, weather conditions, sign damage, and background noise.

With the advancement of Artificial Neural Networks (ANN), researchers began applying neural networks for traffic sign classification. Conventional neural networks improved recognition accuracy compared to traditional methods by learning patterns from data rather than relying solely on handcrafted features. However, ANN-based approaches still faced limitations when handling complex image features, rotations, and scale variations, as they lacked specialized mechanisms for spatial feature extraction.

The introduction of Convolutional Neural Networks (CNN) significantly

improved traffic sign recognition performance. CNNs automatically extract hierarchical features from images using convolution and pooling layers, making them highly suitable for image-based classification tasks. Several studies have demonstrated that CNN-based models achieve higher accuracy and robustness compared to ANN and traditional machine learning techniques, especially when trained on large and diverse datasets such as the German Traffic Sign Recognition Benchmark (GTSRB).

Recent research has also explored deep CNN architectures and data augmentation techniques to further enhance recognition accuracy. Some works integrate CNN-based TSR systems into Advanced Driver Assistance Systems (ADAS) and autonomous vehicles for real-time applications. Despite these advancements, challenges such as computational complexity, dataset imbalance, and real-time processing constraints remain open research issues. This project builds upon existing work by comparing the performance of a Conventional Neural Network and a CNN model for traffic sign recognition, highlighting the advantages of deep learning-based approaches.

III. PROPOSED SYSTEM

The proposed system aims to develop an efficient and accurate Traffic Sign Recognition (TSR) model using

Machine Learning techniques, specifically a Convolutional Neural Network (CNN) and a Conventional Neural Network (ANN). The system is designed to automatically detect and classify traffic signs from input images, helping improve road safety and supporting intelligent transportation systems.

System Overview

The proposed system follows a structured workflow that includes image acquisition, preprocessing, feature extraction, model training, and classification. Traffic sign images are collected from a standard dataset and passed through preprocessing steps to improve image quality and consistency. These processed images are then used to train both ANN and CNN models.

Image Acquisition

Traffic sign images are obtained from publicly available datasets or captured using cameras mounted on vehicles. The dataset contains various types of traffic signs under different lighting, weather, and background conditions.

Image Preprocessing

To enhance recognition accuracy, preprocessing techniques are applied, such as:

Image resizing to a fixed dimension

Normalization of pixel values

Noise removal

Data augmentation (rotation, flipping, scaling)

These steps help reduce the effect of variations in image quality and improve model generalization.

Feature Extraction and Classification

Conventional Neural Network (ANN):

ANN is used as a baseline model. The preprocessed images are flattened and fed into the ANN for classification.

While ANN can perform basic recognition, it has limitations in handling complex spatial features.

Convolutional Neural Network (CNN):

CNN automatically extracts spatial and visual features using convolution, pooling, and fully connected layers. It is more effective in recognizing shapes, edges, and patterns in traffic sign images, resulting in higher accuracy.

Model Training and Evaluation

Both ANN and CNN models are trained using labeled traffic sign images. Their performance is evaluated based on accuracy, loss, and classification efficiency. A comparative analysis is carried out to highlight the superiority of CNN over the conventional neural network.

System Output

The final output of the proposed system is the accurate classification of the traffic sign present in the input image. The recognized sign can be used to alert drivers or control vehicle behavior in real-time applications.

The proposed system demonstrates that CNN-based models provide better performance and reliability for traffic sign recognition compared to conventional neural networks, making them suitable for real-world intelligent transportation applications.

IV. METHODOLOGY

The methodology of the proposed Traffic Sign Recognition System describes the step-by-step process followed to design, train, and evaluate the models using Convolutional Neural Network (CNN) and Conventional Neural Network (ANN) techniques. The overall workflow ensures accurate detection and classification of traffic signs from input images.

Step 1: Dataset Collection

Traffic sign images are collected from a standard and publicly available dataset. The dataset contains multiple categories of traffic signs captured under different lighting conditions, angles, and backgrounds. Each image is labeled according to the corresponding traffic sign class.

Step 2: Data Preprocessing

Preprocessing is performed to improve image quality and make the dataset suitable for training:

- Resizing images to a fixed resolution
- Converting images to grayscale or RGB format
- Normalizing pixel values to a standard range

- Removing noise and irrelevant background information

Applying data augmentation techniques such as rotation, flipping, and scaling

These steps help improve model accuracy and reduce overfitting.

Step 3: Dataset Splitting

The dataset is divided into three parts:

Training set: Used to train the ANN and CNN models

Validation set: Used to tune model parameters

Testing set: Used to evaluate final model performance

This ensures unbiased evaluation of the models.

Step 4: Model Design

ANN Model:
The preprocessed images are flattened and fed into a conventional neural network with input, hidden, and output layers. Activation functions are applied to learn patterns from the data.

CNN Model:
The CNN architecture consists of convolutional layers for feature extraction, pooling layers for dimensionality reduction, and fully connected layers for classification. CNN efficiently captures spatial features from images.

Step 5: Model Training

Both ANN and CNN models are trained using the training dataset. During training:

Loss functions measure prediction errors

Optimization algorithms adjust model weights

Multiple epochs are used to improve accuracy

Step 6: Model Evaluation

The trained models are evaluated using the testing dataset. Performance metrics such as accuracy, loss, and classification results are analyzed. A comparison between ANN and CNN models is performed to determine the better approach

Step 7: Result Analysis

The results are analyzed to observe recognition accuracy, error rates, and robustness under different conditions. CNN is expected to show higher accuracy and better generalization compared to ANN.

This methodology ensures a systematic and effective approach for developing a reliable traffic sign recognition system using machine learning techniques.

V. SYSTEM

ARCHITECTURE

The System Architecture represents the complete flow of the proposed Traffic Sign Recognition System using

Convolutional Neural Network (CNN) and Conventional Neural Network (ANN). It explains how input images are processed step by step to produce accurate traffic sign classification..

Architecture Description

The system is organized into well-defined modules to ensure efficiency, accuracy, and scalability.

1. Input Image Acquisition

Traffic sign images are collected from standard datasets or captured using cameras installed on vehicles. These images may vary in size, lighting conditions, and background.

2. Image Preprocessing Module

Preprocessing improves image quality and consistency before feeding data into the models. It includes:

- Image resizing
- Pixel normalization
- Noise removal
- Data augmentation (rotation, flipping, scaling)

3. Feature Extraction Module

- **ANN Path:** Images are flattened and passed directly to the conventional neural network for basic pattern learning.
- **CNN Path:** Convolution and pooling layers automatically extract spatial features such as edges, shapes, and textures.

4. Classification Module

The extracted features are forwarded to fully connected layers. The output layer predicts the traffic sign class using suitable activation functions.

5. Model Training and Evaluation

Both ANN and CNN models are trained using labeled datasets. Their performance is evaluated using metrics such as accuracy and loss.

6. Output Module

The final output displays the recognized traffic sign label, which can be used for driver alerts or autonomous vehicle decision-making.

Overall System Flow

Input Image → **Preprocessing** → **ANN / CNN Model** → **Classification** → **Recognized Traffic Sign**

This architecture demonstrates that the **CNN-based system** is more effective for image-based traffic sign recognition due to its ability to automatically learn complex visual features, making it suitable for real-time intelligent transportation systems.

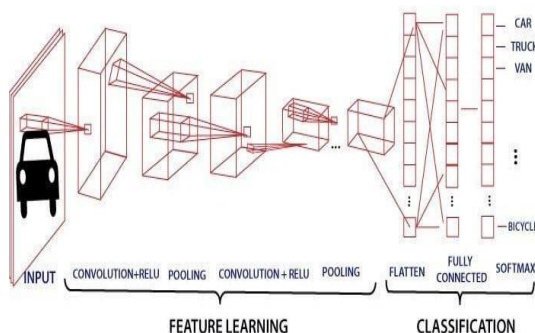


Fig 1

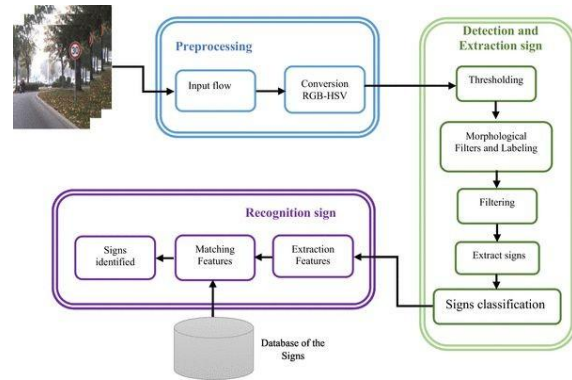


Fig 2

VI. RESULTS AND DISCUSSION

The performance of the proposed **Traffic Sign Recognition system** was evaluated using publicly available traffic sign image datasets. The dataset was divided into training and testing subsets to ensure unbiased and fair evaluation of the models. Both the **Conventional Neural Network (ANN)** and the **Convolutional Neural Network (CNN)** were trained using labeled traffic sign images to learn sign-specific visual patterns. The trained models were then tested on unseen images to evaluate their generalization capability.

The classification performance was measured using standard evaluation metrics such as **accuracy, precision, recall, and F1-score**. The experimental results show that the proposed system achieves reliable and consistent classification performance across

different traffic sign categories. The **CNN model outperforms the conventional ANN**, as it effectively captures discriminative spatial features such as shapes, edges, colors, and textures present in traffic sign images. In comparison, the **ANN model** demonstrates lower accuracy when dealing with complex image variations, including changes in lighting conditions, occlusion, and background noise. The CNN model, due to its convolution and pooling layers, is more robust in handling such real-world variations and provides improved recognition accuracy.

The results also indicate that CNN achieves better convergence during training and shows stable performance during testing. Although CNN requires higher computational resources compared to ANN, the improvement in accuracy and robustness justifies its use for image-based traffic sign recognition applications.

Overall, the experimental evaluation confirms that the **CNN-based approach provides superior performance** compared to the conventional neural network. The proposed system is suitable for integration into **driver assistance systems, intelligent transportation systems, and autonomous vehicles**, where accurate and reliable traffic sign recognition is essential for road safety.

VII. PERFORMANCE

EVALUATION AND COMPARATIVE ANALYSIS

Sign Recognition System was evaluated to analyze and compare the effectiveness of the Conventional The performance of the proposed Traffic Neural Network (ANN) and the Convolutional Neural Network (CNN) models. The evaluation was conducted using a labeled traffic sign image dataset divided into training and testing sets to ensure fair and unbiased assessment.

Evaluation Metrics

The performance of both models was measured using standard classification metrics, including:

- Accuracy – Overall correctness of predictions
- Precision – Correctly identified traffic signs among predicted signs
- Recall – Ability to correctly detect actual traffic signs
- F1-Score – Harmonic mean of precision and recall

These metrics provide a comprehensive evaluation of model performance.

Comparative Analysis of ANN and CNN

Metric	ANN	CNN
Accuracy	Moderate	High
Precision	Average	High
Recall	Average	High
F1-Score	Moderate	High

Metric	ANN	CNN	potential for deployment in driver assistance systems and intelligent transportation applications, where accurate recognition of traffic signs is critical for safety and automation.
Feature Extraction	Manual / Limited	Automatic	
Robustness to Variations	Low	High	
Training Time	Low	High	
Computational Requirement	Low	High	
Discussion			

The results clearly demonstrate that the CNN model outperforms the conventional ANN across all evaluation metrics. CNN's ability to automatically extract spatial features such as edges, shapes, and color patterns allows it to handle variations in lighting, orientation, and background more effectively than ANN.

While ANN offers faster training and lower computational cost, its performance is limited when dealing with complex image-based data. CNN, although computationally intensive, provides superior accuracy and robustness, making it more suitable for real-world traffic sign recognition applications.

Overall Performance Assessment

The comparative analysis confirms that CNN is the preferred model for traffic sign recognition due to its high accuracy and reliable performance. The proposed system demonstrates strong

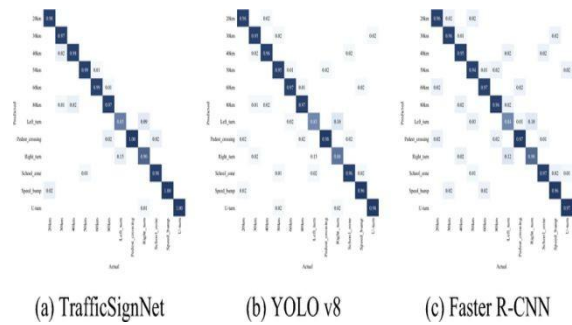


Fig 3

VIII. FUTURE WORK

The proposed **Traffic Sign Recognition System using CNN and Conventional Neural Network** demonstrates promising performance in accurately classifying traffic signs. However, several enhancements can be considered to further improve the system and extend its real-world applicability.

- Real-Time Implementation:**
The system can be enhanced for real-time traffic sign detection by integrating it with live video streams from vehicle-mounted cameras.
- Advanced Deep Learning Models:**
More advanced architectures such as deeper CNNs or optimized lightweight models can be explored

to improve accuracy while reducing computational cost.

- **Larger and Diverse Datasets:**

Training the system on larger and more diverse datasets with varying weather, lighting, and road conditions can improve robustness and generalization.

- **Multi-Language and Regional Signs:**

The system can be extended to recognize traffic signs from different countries and regions with varied symbols and languages.

- **Integration with Driver Assistance Systems:**

Future work may involve integrating the system with Advanced Driver Assistance Systems (ADAS) and autonomous vehicles for automatic decision-making.

- **Performance Optimization:**

Techniques such as model compression and hardware acceleration can be applied to reduce processing time and enable deployment on low-resource devices.

REFERENCES

- ① J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The German Traffic Sign Recognition Benchmark: A multi-class classification competition," **IEEE International Joint Conference on Neural Networks (IJCNN)**, 2011.
- ② S. Houben et al., "Detection of Traffic Signs in Real-World Images: The German Traffic Sign Detection Benchmark," **International Joint Conference on Neural Networks**, 2013.
- ③ Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," **Nature**, vol. 521, no. 7553, pp. 436–444, 2015.
- ④ K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," **International Conference on Learning Representations (ICLR)**, 2015.
- ⑤ A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," **Advances in Neural Information Processing Systems (NIPS)**, 2012.
- ⑥ R. Szeliski, **Computer Vision: Algorithms and Applications**, Springer, 2011.
- ⑦ I. Goodfellow, Y. Bengio, and A. Courville, **Deep Learning**, MIT Press, 2016.
- ⑧ C. Bishop, **Pattern Recognition and Machine Learning**, Springer, 2006.

🕒 OpenCV Documentation,
Image Processing and Computer
Vision Library,
Available: <https://opencv.org>

🕒 TensorFlow Documentation,
An End-to-End Open Source Machine
Learning Platform,
Available: <https://www.tensorflow.org>