Traffic Sign Classification Using Deep Learning

A Srujana
Assistant Professor
Department of CSE(AI&ML)
St. Peter's Engineering College, Hyderabad.
srujana.a@stpetershyd.com

Abstract — **Detection and recognition of traffic signs are very important and could potentially be used for driver assistance to reduce accidents and eventually in driverless automobiles Also traffic signs are essential part of day to day lives. They contain critical information that ensures the safety of all the people. As there are number of traffic signs throughout the world, it is almost impossible for human beings to remember them and identity their meaning which create huge traffic accidents and human loss throughout the world so it is important to establish this project that will remember the traffic signs of all the country throughout the world. Traffic signs classification is the process of identifying which class a traffic sign belongs to. In this project with the help of deep learning, different traffic signs are identified and classified into different categories which helps in reducing various traffic accidents and also reduces human time to remember different traffic signs. In this project, traffic sign recognition using Convolutional Neural Network (CNN) is implemented, the CNN will be trained by using GTSRB dataset of 43 different classes containing 50,000 images of traffic signs the results will show 94% accuracy.**

Keywords— Convolutional Neural Network, Traffic Sign Recognition, Tensor flow, Keras.

I. INTRODUCTION

Traffic sign detection and recognition has gained importance with advances in image process thanks to the advantages that such a system could give. The recent developments and interest in self-driving cars has conjointly exaggerated the interest during this field. A traffic sign detection and recognition system can give the flexibility for good cars and good driving. In self-driving cars, many passengers fully depend on the car for traveling. But to achieve level 5 autonomous, it is necessary for vehicles to understand and follow all traffic rules. In the world of Artificial Intelligence and advancement in technologies, many researchers and big companies like Tesla, Uber etc. are working on autonomous vehicles and self-driving cars. So, for achieving accuracy in this technology, the vehicles should be able to interpret traffic signs and make decisions accordingly. Without traffic signs, all the drivers would be clueless about what might be ahead to them and roads can become mess. The annual global roach crash statistics say that over 3,280 people die every day in a road accident. These numbers would be much higher in case if there were no traffic signs. On the other hand, researchers and big companies are working extensively on proposing solutions to self-driving cars like Tesla, Uber, Google, Audi, BMW, Ford, Toyota, Mercedes, Volvo, Nissan, etc. These autonomous vehicles need to follow the traffic rules and for that, they have to understand the message conveyed through traffic signs.

Traffic Sign Classification is employed to detect and classify traffic signs to inform and warn a driver beforehand to avoid violation of rules. There are certain disadvantages of the existing systems, used for classification, like incorrect predictions, hardware cost and maintenance, which is to a great extent resolved by the proposed system. The proposed approach implements a traffic signs classification algorithm employing a convolutional neural network. Also, it consists of the feature of web cam detection of the traffic sign. This will help the driver to observe the sign close to his / her eyes on the display screen and thus save his/her time in manually checking the traffic sign each time.

Figure 1: Sign Classification

II. RELATED WORK

Recent advancements in machine learning have significantly enhanced the accuracy and efficiency of traffic sign classification. One notable approach involves the use of deep learning models, which excel in recognizing and categorizing various traffic signs by leveraging hierarchical representations of visual data. For instance, the introduction of residual convolutional blocks with dilated skip connections has led to impressive performance metrics, achieving 99.33% accuracy in the German sign recognition benchmark (Kamran et al.). Simultaneously, the application of Long Short-Term Memory (LSTM) networks combined with stacking meta-learners has demonstrated an ability to minimize misclassifications across multiple datasets, showcasing the potential of these classifiers in a real-world context (Atif et al.). Consequently, these innovative machine learning techniques not only improve recognition rates but also play a crucial role in the safety of autonomous driving systems, highlighting their importance in contemporary traffic management solutions. The logistic regression, SVM, and decision trees offer a high detection rate at the medium level. There are two algorithms, namely ANN and the Naïve Bayesian Networks, which perform better at all parameters. These are very expensive to train. There is a major drawback in all the algorithms. The drawback is that these algorithms don't give the same result in all types of environments. They give better results with one type of dataset and poor results with another type of dataset. Algorithms like KNN and SVM give excellent results with small datasets, and algorithms like logistic regression and fuzzy logic systems give good accuracy with raw and unsampled data.

In 2019, Heta Naik and Prashasti Kanika have done their research on various algorithms [4] like Naïve Bayes, Logistic Regression, J48, and Adaboost. Naïve Bayes is among the classification algorithms. This algorithm depends upon the Bayes theorem. Bayles's theorem finds the probability of an event that is occurring is given. The logistic regression algorithm is similar to the linear regression algorithm. The linear regression is used for the prediction or forecasting of the values. The logistic regression is mostly used for the classification task. The J48 algorithm is used to generate a decision tree and is used for the classification problem. The J48 is the extension of the ID3 (Iterative Dichotomies).

J48 is one of the most widely used and extensively analyzed areas in machine learning. This algorithm mainly works on constant and categorical variables. Adaboost is one of the most widely used machine learning algorithms and is mainly developed for binary classification. The algorithm is mainly used to boost the performance of the decision tree. This is also mainly used for the classification of the regression. The Adaboost algorithm uses fraud cases to classify the transactions, which are fraud and non-fraud. From their work, they have concluded that the highest accuracy is obtained for both the adaboost and logistic regression. As they have the same accuracy, the time factor is considered to choose the best algorithm. By considering the time factor, they concluded that the Adaboost algorithm works well to detect credit card fraud.

In 2019, Sahayasakila V, D. Kavya Monisha, Aishwarya, and Sikhakolli Venkata visalakshiswshai Yasaswi have explained the two important algorithmic techniques [8], which are the Whale Optimization Techniques (WOA) and SMOTE (Synthetic Minority Oversampling Techniques). They mainly aimed to improve the convergence speed and solve the data imbalance problem. The class imbalance problem is overcome using the SMOTE technique and the WOA technique. The SMOTE technique discriminates; all the transactions that are synthesized are again re-sampled to check the data accuracy and are optimized using the WOA technique. The algorithm also improves the convergence speed, reliability, and efficiency of the system.

In 2018, Nakanishi Khare and Saad Yunus Sait explained their work [5] on decision trees, random forests, SVM, and logistic regression. They have taken the highly skewed dataset and worked on such a dataset. The performance evaluation is based on accuracy, sensitivity, specificity, and precision. The results indicate that the accuracy for the logistic regression is 97.7%, for decision trees is 95.5%, for random forest is 98.6%, and for SVM classifier is 97.5%. They have concluded that the Random Forest algorithm has the highest accuracy among the other algorithms and is considered the best algorithm to detect fraud. They also concluded that the SVM algorithm has a data imbalance problem and does not give better results to detect credit card fraud.

III. PROPOSED WORK

Input design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So, inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design. Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design.

The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases. Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

The detailed architecture diagram for the credit card fraud detection system is shown in [Figure. 4.] includes many steps from gathering datasets to deploying models and performing analysis based on results. In this model, we take the Kaggle credit card fraud dataset, and pre-processing is to be done for the dataset. Now to prepare the model, we have to split the data into the training data and the testing data. We use the training data to prepare the Random Forest and the Ad boost models. Then we develop both the models. Finally, the accuracy, precision, recall, and F1-score are calculated for all the models. Finally, the comparison of the credit card fraud transactions more accurately.

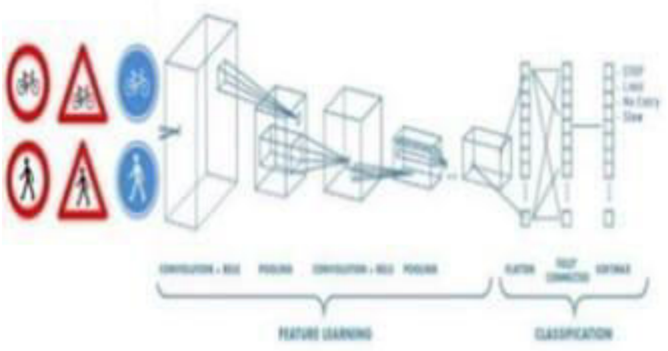

Figure 2: Architectural Diagram

A. CNN Algorithm

A **convolutional neural network** (**CNN**) is a regularized type of feed-forward neural network that learns features by itself via filter (or kernel) optimization. This type of deep learning network has been applied to process and make predictions from many different types of data including text, images and audio.[1] Convolution-based networks are the de- facto standard in deep learning-based approaches to computer vision and image processing, and have only recently have been replaced -- in some cases -- by newer deep learning architectures such as the transformer. Vanishing gradients and exploding gradients, seen during backpropagation in earlier neural networks, are prevented by using regularized weights over fewer connections.[2][3] For example, for *each* neuron in the fully-connected layer, 10,000 weights would be required for processing an image sized 100 × 100 pixels.

However, applying cascaded *convolution* (or cross-correlation) kernels,[4][5] only 25 neurons are required to process 5x5-sized tiles.[6][7]Higher- layer features are extracted from wider context windows, compared to lower-layer features. Then the Random Forest algorithm is an ensemble method. This algorithm is better than the single decision trees because it reduces the overfitting by averaging the result.

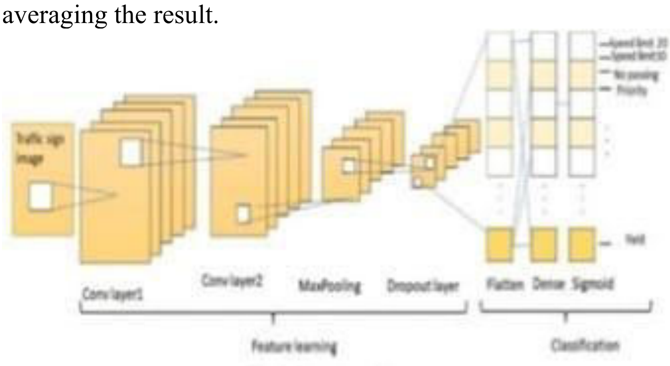

Figure 3: Classification.

Steps for CNN Algorithm:

- **Input layer:** Receives the raw input data, usually images.
- **Convolutional layer:** Applies filters to detect features.
- **Activation layer:** Introduces non-linearity to the system.
- **Pooling layer:** Reduces spatial dimensions, retaining critical information.
- **Flattening:** Converts the pooled feature maps into a vector.
- **Fully connected layer:** Connects every neuron to every neuron in the subsequent layer.

Output layer: Uses a logistic function to perform classification on the output of the final fully connected layer.

IV. EVALUATION AND RESULT ANALYSIS

A. Dataset

In this module, GTSRB is a very famous international database of road signs. In order to train and test the model, some research projects [38] used this database. GTSRB contains 43 kinds of road signs, training and test images taken under real conditions, as shown in Figure 5, a total of more than 50,000 images. In this step, 'train' folder contains 43 folders each representing a different class. The range of the folder is from 0 to 42. With the help of the OS module, iterate over all the classes and append images and their respective labels in the data and labels list.

The PIL library is used to open image content into an array. Stored all the images and their labels into lists (data and labels). Convert the list into NumPy arrays for feeding to the model. The shape of data is (39209, 30, 30, 3) which means that there are 39,209 images of size 30×30 pixels and the last 3 means the data contains colored images (RGB value). With the ski-learn package, use the train test split () method to split training and testing data. From the keras. utils package, use to categorical method to convert the labels present in y train and y test into one-hot encoding.

1. True positive rate, which can be defined as the number of fraudulent transactions that are even classified by the system as fraudulent.
2. True negative rate, which can be defined as the number of legitimate transactions that are even classified as legitimate by the system.
3. A false positive rate, which can be defined as a number of the legal transactions that are wrongly classified as fraud.
4. A false negative rate is defined as the transactions that are fraud but are wrongly classified as legal.

The receiver operating characteristics curve is created by plotting the TPR against the FPR. This can be done at various thresholds. A ROC curve is a graph in which the FPR is the horizontal axis and the TPR is the vertical axis. The graph under the ROC curve is the AUC.

B. Results Analysis

Your dataset represents images of different traffic signs from the German Traffic Sign Recognition Benchmark (GTSRB). It consists of 51,840 images of 43 distinct traffic signs captured under various conditions like different road types, illumination changes, weather conditions, and partial occlusions. The dataset was generated from 10 hours of video recordings in Germany, taken in March, October, and November. These images showcase variations in orientation, shape (triangle, circle, hexagon, diamond), and color combinations. The dataset is split into a 75% training set (39,210 instances) and a 25% test set (12,630 instances) to support machine learning algorithm development.

The confusion matrix [Figure 8] shows us that for the train data, the true positives are 190490 and the false positives are 0, the true negatives are 0 and the false negatives are 330. For the test data, the true positives are 93818 and the false positives are 37; the true negatives are 7 and the false negatives are 125.

The Receiver Operating Characteristic (ROC) is shown in [figure 9] curve depicted in the figure is a graphical representation of the model's performance in binary classification tasks. It plots the True Positive Rate (TPR) (or Recall) against the False Positive Rate (FPR) at various classification thresholds.

Now the dataset is applied for the Adaboost algorithm. The results are obtained similar to those of the Random Forest algorithm.

Accuracy = 0.99907434006830 73

	Precision	recall	F1-score	support
0	0.99938202	0.99969091	0.99953644	93825
1	0.78195489	0.64197531	0.70508475	162

Figure 4: Output for traffic sign classification

The evaluation criteria is shown in [Figure.10] show us that evaluation criteria like precision, recall, and F1-score differ less in the case of the non-fraud cases and differ greatly in those of the fraud cases.

Confusion Matrix on train data

[[190464 120]

[26 210]]

Confusion Matrix on test data

[[93811 65]

[14 97]]

Figure 5:Confusion matrix for traffic Recognition

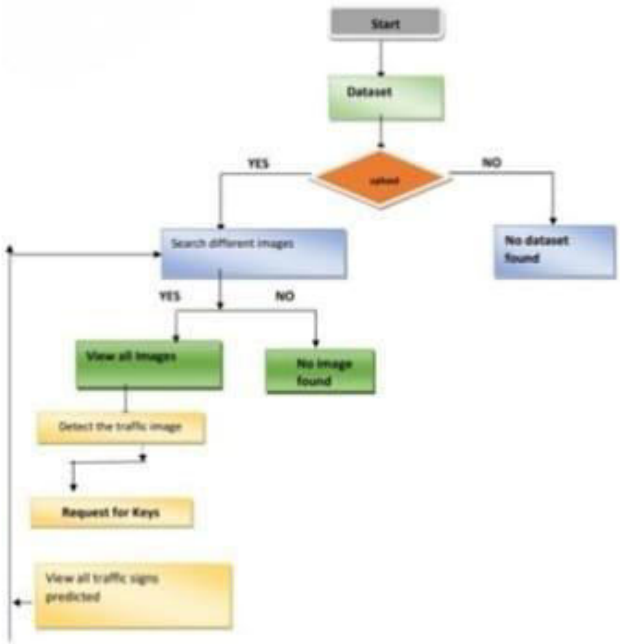

Figure 6: Flow chart

The flow chart in the [figure 6] provides a comparative analysis of the performance metrics—Accuracy, Precision, Recall, and F1-Score—for the convolutional neural network algorithms.

V. TESTING

Testing is a procedure, which uncovers blunders in the program. Programming testing is a basic component of programming quality affirmation and speaks to a definitive audit of determination, outline and coding. The expanding perceivability of programming as a framework component and chaperon costs related with a product disappointment are propelling variables for we arranged, through testing. Testing is the way toward executing a program with the plan of finding a mistake. The plan of tests for programming and other built items can be as trying as the underlying outline of the item itself It is the significant quality measure utilized amid programming improvement.

Amid testing, the program is executed with an arrangement of experiments and the yield of the program for the experiments is assessed to decide whether the program is executing as it is relied upon to perform.

A technique for programming testing coordinates the outline of programming experiments into an all-around arranged arrangement of steps that outcome in fruitful improvement of the product. The procedure gives a guide that portrays the means to be taken, when, and how much exertion, time, assets will be required. Keeping in mind the end goal to ensure that the framework does not have blunders, the distinctive levels of testing techniques that are connected at varying periods of programming improvement are:

Unit Testing is done on singular modules as they are finished and turned out to be executable. It is restricted just to the planner's prerequisites. It centers testing around the capacity or programming module. It Concentrates on the interior preparing rationale and information structures. It is rearranged when a module is composed with high union.

- Reduces the quantity of experiments.
- Allows mistakes to be all the more effectively anticipated and revealed.

VI. CONCLUSION

The Proposed System is designed to detect a method for automatic fine-grained recognition of traffic signs is presented. The classification process is carried out by using a single CNN that alternates convolutional and spatial transformer modules. To find out the best CNN architecture, several empirical experiments are conducted in order to investigate the impact of multiple spatial transformer network configurations within the CNN, together with the effectiveness of four stochastic gradient descent optimization algorithms. The CNN model outperforms all previous state-of-the-art methods and achieves a recognition rate accuracy of 99.71% in the

GTSRB, and it is therefore currently top-1 ranked. Furthermore, our proposed approach needs no hand-crafted data augmentation and jittering used in prior work (Ceresin et al., 2012; Jin et al., 2014; Sermonette & Lacuna, 2011). Moreover, there are fewer memory requirements and the network has a lower number of parameters to learn compared with existing methods since the use of several CNNs in a committee or in an ensemble is avoided. Although our method is ranked in the top positions of the German and Belgian datasets, there have been several recent releases of publicly available traffic sign recognition datasets: these have not yet been tested since they are less established than previous datasets. Nevertheless, to the best of our knowledge, no other scientific paper analyses the use of several STNs and the comparison of stochastic gradient descent optimizers in the traffic sign classification problem domain.

These experiments and their results can help other researchers to apply this new proposal to these new datasets.

REFERENCES

[1]. Statista.com. retail e-commerce revenue forecast from 2017 to 2023 (in billion U.S. dollars). Retrieved April 2020, from India: https://www.statista.com/statistics/280925/e- commercerevenueforecast-in-india/

[2]. Yashvi Jain, Namrata Tiwari, Shripriya Dubey, and Sarika Jain: A Comparative Analysis of Various Credit Card Fraud Detection Techniques, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7 Issue-5S2, January 2019.

[3]. Adi Saputra1, Suharjito2L: Fraud Detection using Machine Learning in e-Commerce, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 9, 2019.

[4]. P.Deepan and L.R. Sudha, "Object Classification of Remote Sensing Image Using Deep Convolutional Neural Network", The Cognitive Approach in Cloud Computing and Internet of Things Technologies for Surveillance Tracking Systems, pp.107-120, 2020. https://doi.org/10.1016/B978-0-12-816385-6.00008-8.

[5]. Ganga Rama Koteswara Rao and R. Satya Prasad, "Shielding the Networks Depending on Linux Servers Against Arp Spoofing, International Journal of Engineering and Technology (UAE), Vol. 7, pp. 75-79, May 2018, ISSN No. 2227-524X, DOI: 10.14419/ijet. v7i2.32.13531.

[6]. Heta Naik, Prashasti Kanikar: Credit card fraud detection based on machine learning algorithms, International Journal of Computer Applications (0975–8887), Volume 182 No. 44, March 2019.

[7]. Navanshu Khare, Saad Yunus Sait: Credit Card Fraud Detection Using Machine Learning Models and Collating Machine Learning Models, International Journal of Pure and Applied Mathematics Volume 118 No. 20 2018, 825-838 ISSN: 1314- 3395.

[8]. Randula Koralage, Faculty of Information Technology, University of Moratuwa, Data Mining Techniques for Credit Card Fraud Detection.

[9]. Roy, Abhimanyu, et al.: Deep learning detecting fraud in credit card transactions, 2018 Systems and Information Engineering Design Symposium (SIEDS), IEEE, 2018.

[10]. P.Deepan and L.R. Sudha, "Scene Classification of Remotely Sensed Images using Ensembled Machine Learning Models", Proceedings in Lecturer Notes on Electrical Engineering, Springer Nature, pp.535-550, 2021, https://doi.org/10.1007/978-981-16-0289-4_39

[11]. Sahayasakila V, D. Kavya Monisha, Aishwarya, Sikhakolli Venkatavisalakshiseshsai Yasaswi: Credit Card Fraud Detection System using Smote Technique and Whale Optimization Algorithm, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958, Volume-8 Issue-5, June 2019.