Vol-06 Issue 01, January -2022 ISSN: 2456-9348
Impact Factor: 5.004

iJETRM
International Journal of Engineering Technology Research & Management

TRAFFIC SIGN CLASSIFICATION USING DEEP LEARNING TECHNIQUES

K. Sayatankar*[1], O. Parvathi*[2], K. Suryakala*[3], G.Sree Lakshmi Siddardha*[4], J. Satya Sri*[5], P.Sai Sampath Kumar*[6]

*[1]Assistant Professor, CSE Dept, Pragati Engineering College(A), Surampalem, Andhra Pradesh, India.

*[2]Assistant. Professor, Department Of CSE, Pragati Engineering College(A), Surampalem, Andhra Pradesh, India.

*[3,4,5,6] B.TECH. III Year Students, CSE Dept, Pragati Engineering College(A), Surampalem, Andhra Pradesh, India.

ABSTRACT

The main aim of our research is to classify traffic signs by using various deep learning algorithms. Traffic sign classification is the process of automatically recognizing traffic signs along the road, including speed limit signs, yield signs, merge signs, etc. In this Convolutional Neural Networks(CNN) algorithm is applied to analyze visual imagery which is a special type of neural network that roughly imitates human vision.

Keywords:

Convolutional Neural Networks(CNN),German Traffic Sign Recognition Benchmark (GTSRB),TensorFlow, OpenCV

INTRODUCTION

Nowadays technologies are furthering our goals and helping with automation in every field making humans feel effortless, comfortable and secure .As a human is prone to making mistakes, a machine in his/her place would certainly be more efficient, both in terms of speed and accuracy. This research is very helpful in designing self- driving cars. They will be able to perceive traffic signs and react according to the input received. With the model that we have developed ,self-driving cars can easily classify traffic signs so that can obviously reduce the chance of occurring accidents. In this research, we have built a deep neural network model using Convolutional Neural Networks that have the capability to classify traffic signals that are present in the image based on its class. With the model that awe have developed, we were able to detect as well as classify traffic signals, which is very crucial to self-driving cars because it can otherwise lead to fatal accidents.

Convolutional Neural Networks

Convolutional neural networks are deep learning algorithms that are very powerful for the analysis of images. CNN is a powerful algorithm for image processing. These algorithms are currently the best algorithms we have for the automated processing of images. Many companies use these algorithms to do things like identifying the objects in an image. Images contain data of RGB combination. Matplotlib can be used to import an image into memory from a file. The computer doesn't see an image, all it sees is an array of numbers. Colour images are stored in 3-dimensional arrays. The first two dimensions correspond to the height and width of the image (the number of pixels). The last dimension corresponds to the red, green, and blue colours present in each pixel.

Layers Of CNN: Convolutional Neural Networks specialized for applications in image & video recognition. CNN is mainly used in image analysis tasks like Image recognition, Object detection & Segmentation.

There are three types of layers in Convolutional Neural Network.

1. Convolutional Layer
2. Pooling Layer
3. Fully-Connected layer

iJETRM
International Journal of Engineering Technology Research & Management

Convolutional Layer: In a typical neural network each input neuron is connected to the next hidden layer. In CNN, only a small region of the input layer neurons connect to the neuron hidden layer.
Pooling Layer: The pooling layer is used to reduce the dimensionality of the feature map. There will be multiple activation & pooling layers inside the hidden layer of the CNN.
Fully-Connected layer: Fully Connected Layers form the last few layers in the network. The input to the fully connected layer is the output from the final Pooling or Convolutional Layer, which is flattened and then fed into the fully-connected layer.

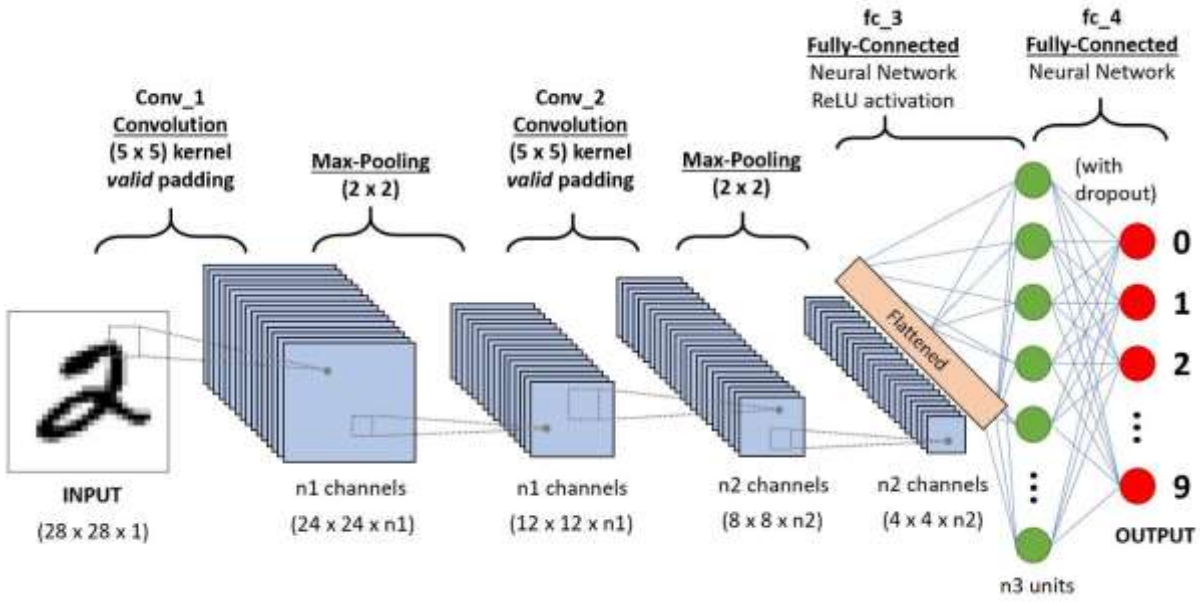

Figure 1:CNN

REQUIREMENTS USED

We have used multiple dependencies for our research. As this research requires manipulation as well as visualization of the data being used, there was extensive use of libraries. The main dependencies we used for our research are:

1. TensorFlow
2. Keras
3. Numpy
4. Pandas
5. Opencv
6. Matplotlib
7. Tkinter
8. Sci-kit learn

TensorFlow: TensorFlow is an open-source software library for high performance numerical computation. Its flexible architecture allows easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices. Originally developed by researchers and engineers from the Google Brain team within Google's AI organization, it comes with strong support for machine learning and deep learning and the flexible numerical computation core is used across many other scientific domains.
NumPy: NumPy is the fundamental package for scientific computing with Python. It contains among other things:

- a powerful N-dimensional array object
- sophisticated (broadcasting) functions

iJETRM
International Journal of Engineering Technology Research & Management

- tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities
- Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases

OpenCV: OpenCV (Open-Source Computer Vision Library) is released under a BSD license and hence it's free for both academic and commercial use. It has C++, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform.

Pandas :pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself. Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007 to 2010.

Scikit-learn: Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

Matplotlib: Matplotlib is a low-level graph plotting library in python that serves as a visualization utility. Matplotlib was created by John D. Hunter. Matplotlib is open source and we can use it freely. Matplotlib is mostly written in python, a few segments are written in C, Objective-C and JavaScript for Platform compatibility

Tkinter: The tkinter package ("Tk interface") is the standard Python interface to the Tcl/Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, including macOS, as well as on Windows systems. Running python -m tkinter from the command line should open a window demonstrating a simple Tk interface, letting you know that tkinter is properly installed on your system, and also showing what version of Tcl/Tk is installed, so you can read the Tcl/Tk documentation specific to that version.

DATASET

The dataset we have used for our research was the GTSRB- German Traffic Sign Recognition Benchmark. This is a very well-known dataset for traffic signs in websites like Kaggle. This dataset can be divided into 43 classes which consists of 50000 images for training, validation and testing purposes . We have divided the data set into training, validation and testing set, which further helped us in understanding how well our architecture was working.

iJETRM
International Journal of Engineering Technology Research & Management

Figure 2:Images In The Dataset

MOTIVATION

- In the past and recent times, there have been many road accidents where the main reason for these being inadequate knowledge of road and traffic signs.
- Even though speed is one of the key issues for the cause of such atrocities, in a survey, it was found out that the second most heard reason was an individual not knowing what a particular traffic sign meant .
- We strongly believe that the research that we have done would help individuals learn these signs intuitively, especially the adolescence of 21st century, who also stay and live around technology, which is growing faster than ever .
- Our research focuses on classifying traffic signs, provided an image to it through deep learning, image processing through OpenCV and a CNN is having been developed in Python GUI using Tkinter library.

CONCLUSION

The classification of traffic signs using CNN with different filters helped improve the system by selecting the most accurate model with an accuracy of 96% and train the model using GTRSB dataset and test it with Indian Dataset. In this research using TensorFlow, CNN and OpenCV, we have successfully developed a traffic sign classifier which attained an accuracy of 96%, which is functioning better than many other models that have been developed from other researches. We also developed a python GUI which looks interactive and intuitive to use, which takes an image as input and presents the predicted traffic sign to the user

FUTURE SCOPE

As a part of future work, we were planning on integrating our model into a real time camera, which would further improve its functionality and application. This can further be included in industrial level products such as driverless cars in the future, provided we integrate our research work into a real time system. Another direction for further research is to develop a real time traffic sign recognition system which captures a video by a camera mounted on the vehicle, detects and recognize the traffic signs in real time and gives the result to the driver within a sufficient time frame in order to take the right action. More research can be done to make an AI that can verify reports ,images and videos automatically without human invention and makes the entire process smooth and fast.

iJETRM
International Journal of Engineering Technology Research & Management

REFERENCES

[1].German Traffic Sign Recognition Benchmark https://www.kaggle.com/meowmeowmeowmeo wmeow/gtsrb-german-traffic-sign
[2]. J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The German traffic sign recognition benchmark: a multi-class classification competition,". in Proc. IEEE IJCNN, 2011, pp. 1453–1460.
[3]. Valentyn Sichkar, Sergey A. Kolyubin, ""Effect of various dimension convolutional layer filters on traffic sign classification accuracy".
[4]. Yi Yang, Hengliang Luo, Huarong Xu and Fuchao Wu, Towards Real-Time Traffic Sign Detection and Classification 2014, IEEE.
[5]. Kai Li, Weiyao Lan, "Traffic indication symbols recognition with shape context". Department of Automation Xiamen University, China, 2011, IEEE [9]. DarkNet Framework https://pjreddie.com/darknet
[6]Xu, H., & Srivastava, G. (2020). Automatic recognition algorithm of traffic signs based on convolution neural network. Multimedia Tools and Applications, 1-15
[7]Yakimov PY (2015) Preprocessing digital images for quickly and reliably detecting road signs. Pattern Recognition of Image Anal 25(4):729–732.