Traffic Sign Classification using Convolutional Neural Networks

COMP 6721 Applied Artificial Intelligence (Fall 2022)

Group - H

1. Problem Statement and Application

Traffic sign classification focuses on identifying the kind of traffic sign in the given image/frames of images. There are various reasons why this problem is important. One being as we are entering into a world of Autonomous driving! The vehicle should be able to detect and understand the meaning of traffic signs and follow them accordingly [1][2]. One other could be applications where we intend to increase driver's focus. If a driver misses a sign, Traffic Sign Recognition (TSR) can make them aware of it so they can react accordingly [3].

Some of the major challenges associated with this application are the changes in illumination in the captured sign images and the fact that the standard priority signs adopted internationally differ in shape, color, and border [4]. We have tried to address these issues by choosing datasets having signs of different countries and by artificially changing illumination in these images for training.

A convolutional neural network (CNN) is a type of artificial neural network used primarily for image recognition and processing, due to its ability to recognize patterns in images. The goal behind this application is to compare the performance of different CNN models in classification of different traffic signs, analyze the effects of choosing different hyperparameters and choose a final model as per these evaluations.

2. Image Dataset Selection

The chosen datasets have images of signs of different countries. Further these datasets combined consist of signs of more than 250 different classes like stop sign, speed limit, school ahead, pedestrian crossing etc. The images also differ from each other in terms of their colour, shape, borders etc. Datasets were meticulously selected to make sure the datasets do have these properties, as these play a crucial role in addressing the associated challenges mentioned before.

These datasets were found on www.kaggle.com and are accessible/downloadable through the links available in the table as well in the references:

Statistical details of the datasets:

	Number of Images	Image Size	Number of Classes
Dataset 1 [5]	97426	Max: (1496, 974) Min: (22, 44)	205
Dataset 2 [6]	4438	Max: (4608, 3456) Min: (42, 50)	85
Dataset 3 [7]	14405	Max: (3192, 1400) Min: (13, 11)	43

Table 1: Traffic Sign Classification using CNN Dataset

Potentially, the datasets might be modified for better classification. As the classes are imbalanced, appropriate steps would be taken to balance them like by using class weights, resampling, Synthetic Minority Oversampling TEchnique (SMOTE) [8] etc.

3. Possible Methodology

Firstly, we will analyse and pre-process the images. Pre-processing will include resizing of images to appropriate sizes, normalization, resampling to handle class imbalance, changing illuminations etc.

Once pre-processing is done, we will pick 3 architectures from the pre-existing architectures like VGG-16 [9] [10], ResNet50 [11], Inceptionv3, Efficient-Net etc and train them against each of the 3 chosen datasets to get a baseline performance. By doing so, we get 9 models in total.

Next, we would study the effects of changing different hyperparameters in these models and attempt to improve the performance. Hyperparameter considered for tuning would be the loss function, optimizer, learning rate, batch size, and epochs.

Models obtained during each of the above phases would be evaluated and compared to each other using the metrics F1 score and Area Under Curve (AUC) score [12]. Other metrics might be included for comparisons if required.

Novelty in our approach being the detailed analysis of performance of different models for the classification problem and the effects of being trained on different datasets with images of varying shape, size and classes.

We expect the performance of the models to vary as per the training dataset and will choose the best model for the classification, as per the evaluations.

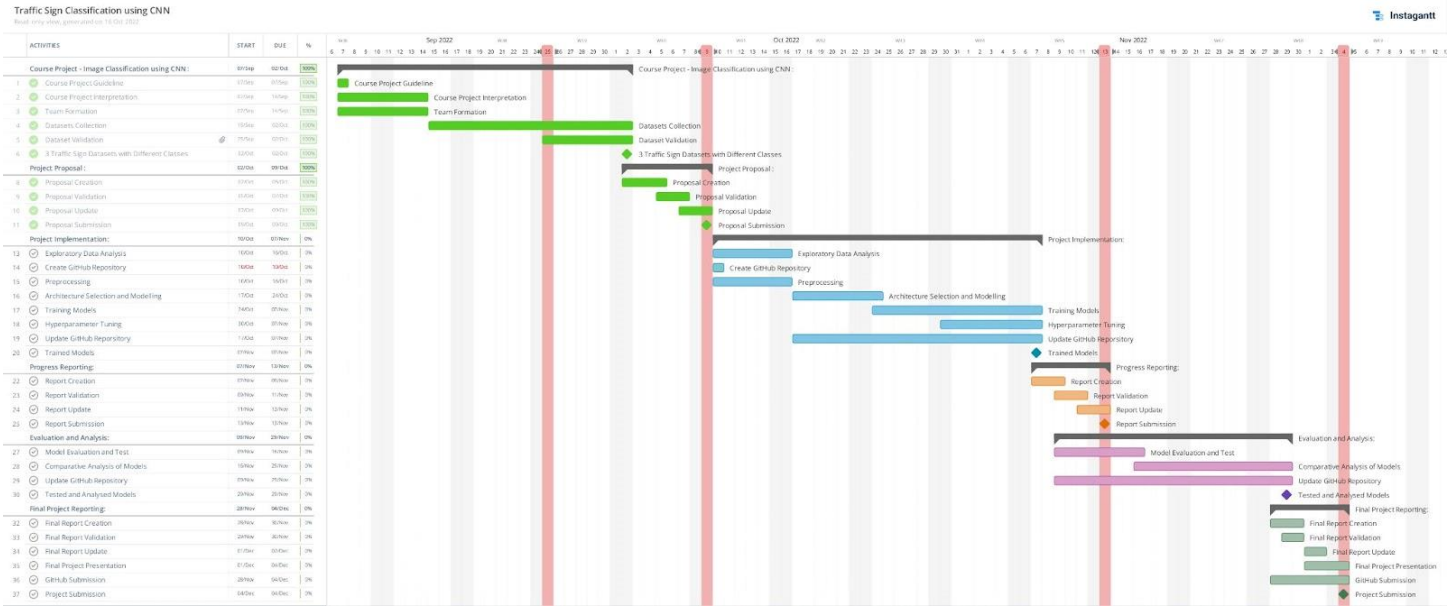

Figure 1: Traffic Sign Classification using CNN Gantt Chart[1]

The above Gantt Chart illustrates the timeline from the initial phase of Crouse Project Guideline to the final phase of Project Submission.

Each section is colour coded for better visualization. The horizontal bar represents the timeline of each task in the sections. The diamond represents milestones as every phase deliverable that we already have and have to further achieve by implementing each phase.

Each milestone must be met on the deadline day which we have shown by the red vertical bar on the Gantt.

We have divided our Project Phases in six-different sections on the Gantt which illustrates different tasks in each section.

Project Phase:

1. Course Project - Image Classification using CNN
2. Project Proposal
3. Project Implementation
4. Progress Reporting
5. Evaluation and Analysis
6. Final Reporting

Our plan is to implement divide and conquer technique. As we are a team of 4 members, we will form 2 groups of 2 members each.

We have already finished with the Project Proposal Phase. We will start with our Project Implementation phase which is the most important step of our classification project.

This phase consists of main tasks:

1. Exploratory Data Analysis
2. Architecture Selection and Modelling

In this phase, 2 members of each team will be working on Exploratory Data Analysis and other 2 members on Architecture Selection and Modelling.

Once, we have analysed our data and selected architectures in order to implement 9 models we plan to individually implement 2/3 models each team member. During this phase we also need to focus and validate our models by performing hyper-parameter tuning in order to analyse and evaluate our models based on different parameters.

We plan to perform this task by discussing some general and custom hyper-parameters based on different models and evaluate our models individually.

After training and tuning all the models we will be focusing on the Evaluation and Analysis phase where we will perform comparative study of the models together and deduce conclusions. This we plan to achieve by working in a team together to get everyone's inputs and have a better understanding of models.

All the reporting and documentation phases are also done using divide and conquer style where we are dividing our tasks individually and then discuss, update and submit out proposals or reports carefully before the deadline.

[1] https://drive.google.com/file/d/1h6QgHGBVrzedCmHS_yIWRGak5BPvZHGX/view?usp=sharing

References

[1] "Traffic Sign Recognition (TSR)." Car Rental Gateway,https://www.carrentalgateway.com/glossary/traffic-sign-recognition/#:~:text=TSR%20is%20also%20important%20for,detection%20faster%20and%20more%20accurate.

[2] Junzhou, Chen, et al. https://www.researchgate.net/publication/354847144_A_real-time_and_high-precision_method_for_small_traffic-signs_recognition.

[3] Seyedjamal, Zabihi. *Detection and Recognition of Traffic Signs Inside the Attentional Visual Field of Drivers*. The University of Western Ontario, 3 Oct. 2017, https://ir.lib.uwo.ca/cgi/viewcontent.cgi?article=6143&context=etd.

[4] *EUROPEAN ROAD SIGNS*. https://www.autoeurope.ca/travel-tips/road-signs/#:~:text=The%20standard%20priority%20signs%20adopted,white%20background%20and%20red%20border.

[5] DELTSOV, DANIIL. TrafficSigns(GTSRB plus 162 Custom Classes)- Dataset 1. https://www.kaggle.com/datasets/daniildeltsov/traffic-signs-gtsrb-plus-162-custom-classes.

[6] DILIP JODH, SARANG. Indian Traffic Signs Prediction (85 Classes), DILIP JODH, SARANG. Indian Traffic Signs Prediction (85 Classes), https://www.kaggle.com/datasets/sarangdilipjodh/indian-traffic-signs-prediction85-classes.

[7] PARSASERESHT, SARA. Persian Traffic Sign Dataset (PTSD), https://www.kaggle.com/datasets/saraparsaseresht/persian-traffic-sign-dataset-ptsd.

[8] Brownle, Jason. "SMOTE for Imbalanced Classification with Python." *Machine Learning Mastery*, 17 Mar. 2021, https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/.

[9] Gaudenz, Boesch. "VGG Very Deep Convolutional Networks (VGGNet) – What You Need to Know Read More at: Https://Viso.Ai/Deep-Learning/Vgg-Very-Deep-Convolutional-Networks/." *VGG Very Deep Convolutional Networks (VGGNet) – What You Need to Know Read More at: https://Viso.Ai/Deep-Learning/Vgg-Very-Deep-Convolutional-Networks/*, https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/#:~:text=and%20many%20more.-,What%20is%20VGG%3F,16%20and%2019%20convolutional%20layers.

[10] Kumar, Abhishek. *VGG-Architecture*. https://www.kaggle.com/code/uniabhi/vgg-architecture.

[11] Gaudenz, Boesch. "Deep Residual Networks (ResNet, ResNet50) – 2022 Guide Read More at: Https://Viso.Ai/Deep-Learning/Resnet-Residual-Neural-Network/." *Deep Residual Networks (ResNet, ResNet50) – 2022 Guide Read More at: Https://Viso.Ai/Deep-Learning/Resnet-Residual-Neural-Network/*, Deep Residual Networks (ResNet, ResNet50) – 2022 Guide Read more at: https://viso.ai/deep-learning/resnet-residual-neural-network/.

[12] Brownlee, Jason. "What Is a Confusion Matrix in Machine Learning." *Machine Learning Mastery*, 18 Nov. 2016.