Semester Project Report
Neural Networks (AI)

Team Members: David van Wuijkhuijse (s5592968), Marcus Harald Olof Persson (s5343798), Richard

University of Groningen

October 31, 2025

Abstract

We implemented a feed-forward convolutional neural network from the group up including loss, activations, optimizer, and LR scheduler. We trained and evaluated the model on the German Traffic Sign Recognition Benchmark (GTSRB) dataset with 38 classes. Our model achieved a test accuracy of xx%.

Contents

Figure 1: The same sign from different distances (and thus sizes) and slightly different angles.

Figure 2: Three different signs of the same type exhibiting variation in lighting conditions and angles.

1   Introduction

State your project task (given or self-chosen). Briefly describe dataset(s) and objectives. Keep it short, direct, and specific to your project.

2   Data

The German Traffic Sign Recognition Benchmark (GTSRB) dataset consists of 51,839 images of traffic signs on german roads, categorized into 43 different classes. The images are taken from dashcam footage of cars driving on the road and thus past the signs. This means that each sign is captured multiple times as the car passes the sign. This lessens the need for data augmentation as each sign is already represented multiple times. Signs are captured in a range of different lighting and weather conditions, as well as from different angles and distances.

The dataset is provided by Stallkamp et al. (2012) [1] [2] from the Institute for Neuroinformatics at the University of Bochum in Germany. By default, the dataset is split into a training set and a testing set. This was originally for a competition at the International Joint Conference on Neural Networks in 2011. Considering we are (considerably) past the deadline for submission, we have chosen to merge the training and testing sets at download time and then create our own split. This allows us to modify the split ratio as we like.

The images themselves are presented as PPM files with varying resolution and aspect ratio depending on the distance and angle of the sign to the camera. The aspect ratio is mostly close to 1:1. For more detailed analysis of the images sizes see Table 1 and Figure 3. There are three color channels (RGB) and

Statistic	Width (px)	Height (px)	Aspect Ratio (W/H)
Mean	50.76	50.34	1.0056
Std	24.50	23.26	0.0734
Min	25	25	0.3681
25th Percentile	34	35	0.9706
50th Percentile (Median)	43	43	1.0000
75th Percentile	58	58	1.0385
Max	266	232	1.4375

Table 1: Descriptive statistics of image dimensions and aspect ratios in the GTSRB dataset.
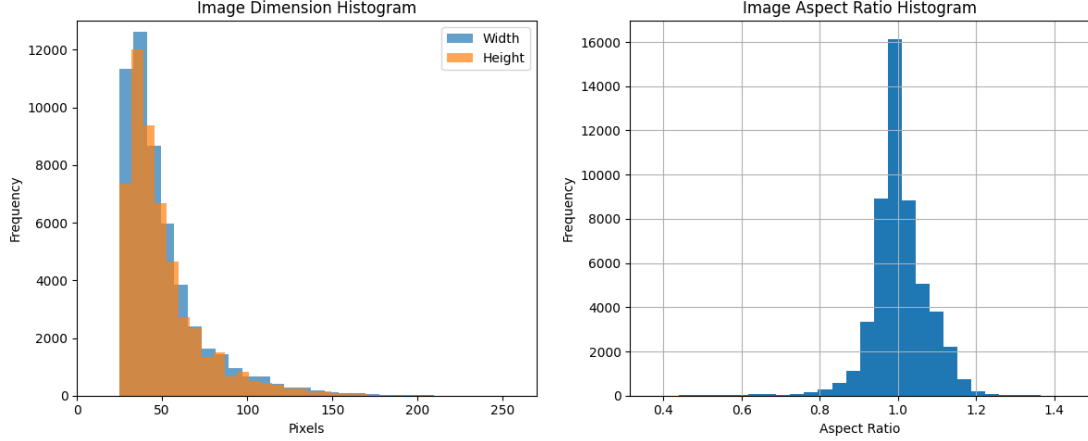

Figure 3: Distribution of image dimensions and aspect ratios in the GTSRB dataset. Most images are around 30-70 pixels in both dimensions, with a long tail towards larger sizes.

the pixel values are in the range of 0-255. See Figure 4 for the distribution of pixel values. The labels for the images are provided in one master file `labels.csv` for the entire dataset. Each line contains the filename, class ID, and bounding box coordinates for the sign in the image.

We load the data as a numpy array of shape (H, W, C) for each image, where H is height, W is width, and C is the color channel (so three). The labels are loaded as integers corresponding to the class IDs. Before feeding the images to the model, we normalize the pixel values to the range [0, 1] by dividing by 255.

- Source of dataset

- Size and type (tabular, images, time series...)

- Key characteristics (with plots or descriptive stats)

- Any preprocessing or cleaning

3   Methods and Experiments

3.1   Pipeline Overview

Diagram or list of your full pipeline: preprocessing, model, optimization.

3.2   Model Description

- Architecture (MLP, CNN, etc.), with math or pseudocode if needed. - Loss function, optimizer, regularization.
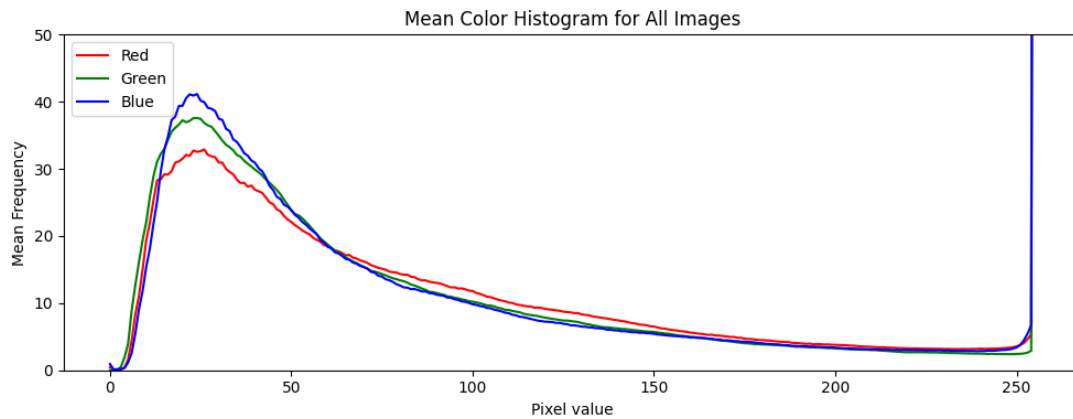
Figure 4: Histogram of pixel values across all images and color channels in the GTSRB dataset. The graph is cut off at frequency=50, but the frequency of flat white is very high due to the reflective nature of signs as well as the common use of the color white, causing (parts of) signs to max out the camera's sensor.

3.3 Training Procedure

- Cross-validation setup, train/test split. - Hyperparameters and how you chose them.

4 Results

- Performance metrics (accuracy, MSE, etc.)

- Learning curves or confusion matrices

- Compare with baseline(s)

5 Discussion

- What worked well and why

- What didn't work, limitations

- Possible improvements or future work

6 Conclusion

Wrap up: summarize objectives, approach, key findings, and lessons learned.

References

[1] Stallkamp, J., Schlipsing, M., Salmen, J., and Igel, C. (2011). The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *IEEE International Joint Conference on Neural Networks*, pages 1453–1460.

[2] Stallkamp, J., Schlipsing, M., Salmen, J., and Igel, C. (2012). Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks*, (0):–.

A Use of AI Tools

Briefly document how (if at all) ChatGPT or other LLMs were used in writing or coding.

B Additional Figures / Materials

Include extra plots, tables, or examples here if needed.