POVa - Traffic sign detection and recognition

Vít Tlustoš (xtlust05), Jiří Vlasák (xvlasa15), Josef Kotoun (xkotou06)

21.12.2023

1 Introduction

Our project is focused on recognizing traffic signs using data from the *Mapillary Traffic Sign Dataset* [1]. Our main focus was on fine-tuning the YOLOv8 model, which tends to produce state-of-the-art results for many object detection tasks in real time. In principle, we employ three different approaches. The first approach involves a one-step process, utilizing a YOLOv8 [2] model for simultaneous traffic sign detection and classification. The second approach employs two separate YOLOv8 models — one for binary detection (sign/no-sign) and another for classification of the pre-detected sign. The third approach involves fine-tuning the Object detection transformer DETR.

2 Data

The *Mapillary Traffic Sign Dataset* is a large-scale, diverse collection of traffic signs captured in real-world settings at street level. Our reasoning for the selection of this dataset is described in the subsequent chapter.

2.1 Annotations

The annotation process involved 15 specially trained experts. To ensure high-quality annotations, the authors consistently monitored the process. Each image required verification by at least two annotators. Furthermore, to confirm the accuracy of these annotations, the authors conducted additional labelling on a smaller overlapping subset and cross-validated the outcomes.

2.2 Analysis

Table 1 presents the number of signs as partitioned into the training, development and testing subsets. Additionally, we have examined the distribution of traffic signs within the dataset, as depicted by Figure 1, and identified a substantial imbalance. Some signs are quite prevalent, appearing in thousands of instances, while others are rare, occurring only a few times, making the task challenging. For example, the *pedestrians-crossing* sign appears 3,933 times, while the *detour-left* sign occurs just 15 times. Additionally, we have thoroughly examined the dataset paper [**?**], leading to the following conclusions:

- Traffic Sign Variety: Mapillary's taxonomy includes over 300 traffic sign categories.
- Global Representation: The dataset encompasses traffic signs from various countries and regions.

- Diverse Capture Devices: Images are sourced using a variety of devices, ranging from low-end smartphones to professional cameras.
- Varying Resolutions: The dataset includes images of different resolutions.
- Diversity of Conditions: Traffic signs are captured under various environmental conditions (sunny, foggy or rainy).

Subset	**Number of Images**
Train	36 589
Dev	5 320
Test	10 544

Table 1: Distribution of signs in the Mapillary Traffic Sign Dataset

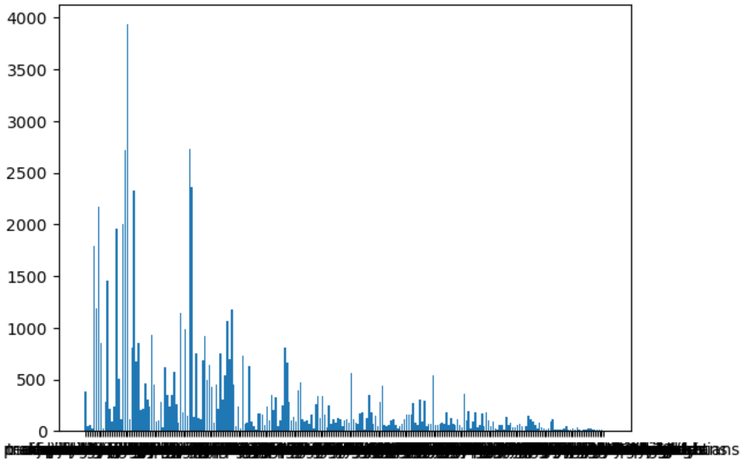

Figure 1: Distribution of traffic signs within the dataset.

2.3 Pre-processing

For the dataset to be usable in our settings (YOLOv8, DETR) the dataset must be per-processed. The data pre-processing scripts are located within the *./notebooks/data.ipynb* file.

2.3.1 Data Acquisition

The data can be downloaded free of charge from the official website. We expect the data to be downloaded and unzipped into the following structure:

```
data/
├── images/
└── annotations/
```

2.3.2 Data Preparation

The user can select whether to create data for simultaneous traffic sign detection and classification or decoupled detection and classification. In the case of simultaneous detection and classification, the following directory containing annotations in YOLOv8 object detection dataset format using the full Mapillary's taxonomy will be created.

```
data/
└─ yolov8/
   └─ detect/
      ├─ train/
      ├─ val/
      └─ datataset.yaml
```

Similarly to simultaneous detection and classification, the *detect/* will be created. In this case, the annotations will feature a single label category to facilitate binary detection. Additionally, directory *classify/* will be created. This directory will contain cut-outs representing a single sign extracted from the image (possibly containing multiple signs). These cut-outs are partitioned based on the YOLOv8 classification dataset format.

```
data/
└─ yolov8/
   └─ classify/
      ├─ train/
      └─ val/
```

2.3.3 Format Conversion

Since the DETR model requires the data to be in the COCO format we use a library called Globox — Object Detection Toolbox for this purpose. This library takes the per-processed dataset in the YOLOv8 format and transforms it into the MS COCO format.

3 Experimental Setup

In our study, we explored three distinct methodologies for traffic sign detection and classification. Initially, we employed the medium variant of the *YOLOv8-m* model for simultaneous traffic sign detection and classification. Additionally, we decoupled the detection and classification tasks by employing a two-model ensemble: one for binary detection of traffic signs and another for classifying these signs using the dataset's complete taxonomy. We hypothesized that decupling detection and classification would result in better results. Finally, the third approach involved fine-tuning the Object Detection Transformer (DETR). We hypothesized that it could outperform the other approaches.

3.1 Simultaneous detection and classification

For the simultaneous traffic detection and classification (one-step) approach, we used the *YOLOv8-m* model. Additionally, we explored the impact of input image resolution on the model performance. We have utilized three distinct input image resolutions, specifically 640, 1280 and 1920. The models were trained for 50 epochs with a batch size of 16 and a learning rate set to 0.01.

3.2 Decoupled detection and classification

For the decoupled detection and classification approach, we used a combination of binary detector and traffic sign classifier. This approach worked well in the Mapillary Traffic Sign Dataset [**?**]. Multiple variants of models for binary detection and classification were trained. The models were trained for 50 epochs with a batch size of 16 and a learning rate set to 0.01.

3.2.1 Binary detection

For binary detection, the *YOLOv8-n*, *YOLOv8-s* and *YOLOv8-m* variants were trained. The *YOLOv8-n* was trained on both 640 and 1280 resolution settings to explore the impact of image resolution on the detection task. The models were trained using a mapillary dataset, but classes of all objects were set to a single class called *traffic-sign*. The models were trained for 50 epochs with a batch size of 16 and a learning rate set to 0.01.

3.2.2 Classification

For Classification, the *YOLOv8-n* and *YOLOv8-m* classifiers were evaluated as well as a simple CNN classifier proposed in the Mapillary Traffic Sign Dataset.

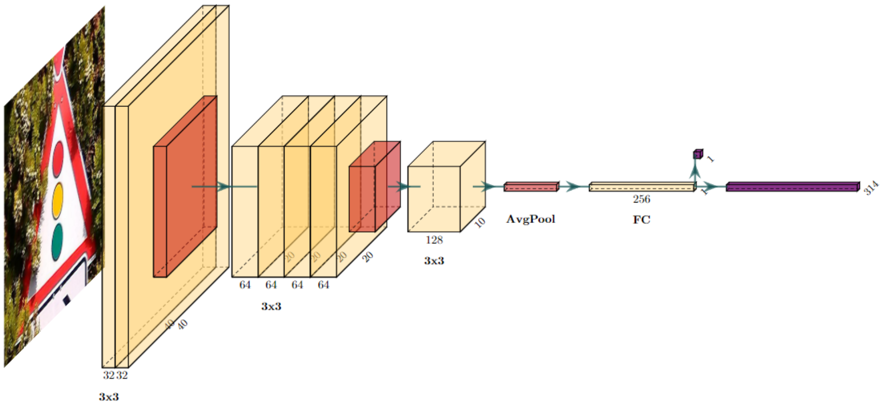

Figure 2: shows the architecture of the simple CNN classifier.

The simple CNN classifier used 7 convolutional layers with 2x2 max pooling and an input size of 40x40. The architecture is illustrated in figure 2. It was trained for 10 epochs with a batch size of 512. This large batch size was chosen so that more traffic sign classes were present in the batch. The learning rate was set to 0.001 and the Adam optimizer was used.

The YOLO classifiers were trained for 100 epochs with a batch size of 512. The learning rate was set to 0.01 and the AdamW optimizer was used. Both networks used an input size of 64x64. A larger input size of 128x128 was also evaluated, but it did not improve the results.

3.3 DETR

We have tried to fine-tune the DETR model via the script located at *notebooks/models.ipynb*. However, we have not managed to make the model converge. We have tried a variety of hyperparameter settings including batch sizes ranging from 4 to 12, gradient accumulation steps ranging from 1 to 64, and learning rates ranging from $1e^{-5}$ to $1e^{-4}$.

4 Evaluation

4.1 Simultaneous approach evaluation

The results of one-step object detection and classification are shown in table 4.1. The medium version of *YOLOv8* was used and the results show, that it performs better on higher-resolution input images.

Model	Input image resolution	Precission	Recall	mAP
YOLOv8-medium	800	62.7%	58.9%	52.3%
YOLOv8-medium	1280	68%	73%	63%
YOLOv8-medium	1920	69%	79%	67.1%

Table 2: Results of two-step object detection and classification.

4.2 Decoupled approach evaluation

4.2.1 Binary detection

The results of binary detection models are shown in table 4.2.1. The results show, that increasing the image resolution of the input image significantly improves the model performance, probably because the traffic signs in images are often very small, so the model is not able to detect the traffic signs in lower-resolution images. The larger model with the same resolution shows some improvement, but not that large in comparison to higher resolution, even though the larger model has almost 4 times more parameters.

Model	Input image resolution	Precission	Recall	mAP
YOLOv8-nano	640	76.1%	73.7%	57%
YOLOv8-nano	1280	79.4%	84.1%	70.5%
YOLOv8-small	1280	82.1%	85.9%	73.9%
YOLOv8-medium	1920	83.9%	87.2%	77.3%

Table 3: shows results of binary detection models on the validation set of the Mapillary Traffic Sign Dataset.

4.2.2 Classification

The results of the classification models on the validation set of the Mapillary Traffic Sign Dataset are shown in table 4.2.2. The *YOLOv8-cls-n* and simple CNN classifiers had similar results despite the *YOLOv8-cls-n* having a much larger number of parameters. The *YOLOv8-cls-m* classifier had the best results, but it also had the largest number of parameters.

Model	Validation accuracy
YOLOv8-n	89.3%
YOLOv8-m	93.1%
Simple CNN	89.5%

Table 4: Results of classification models on the validation set of the Mapillary Traffic Sign Dataset.

4.2.3 Combined

The results of two-step object detection and classification are shown in table 4.2.3. The medium *YOLOv8* with 1920 input resolution was chosen for this task since it performs best. It was evaluated in combination with both *YOLOv8* medium classifier and a simple CNN classifier.

Binary detector model	Input image resolution	Classifier model	mAP
YOLOv8-medium	1920	YOLOv8-medium	66%
YOLOv8-medium	1920	Simple CNN	59.7%

Table 5: Results of two-step object detection and classification.

5 Conclusion

This project explored the use of *YOLOv8* and *DETR* detection models on the *Mapillary Traffic Sign Dataset*. We employed single-stage (simultaneous detection and classification) as well as two-stage (decoupled detection and classification) methods with results favouring the simultaneous approach by a small margin. Additionally, we found that increasing the resolution of input images notably enhances the performance of the models. Lastly, we tried to fine-tune the DETR model but weren't able to make the model converge.

6 Acknoweledgemnts and Links

- The GitHub repository is accessible at https://github.com/vtlustos/pova-traffic-sign-recognition.git
- The fine-tuned models are available at https://huggingface.co/jkot/pova-traffic-sign-recognition-models
- The code for the DETR was adapted from: https://colab.research.google.com/github/woctezuma/finetune-detr/blob/master/finetune_detr.ipynb

References

[1] C. Ertler, J. Mislej, T. Ollmann, L. Porzi, G. Neuhold, and Y. Kuang. The mapillary traffic sign dataset for detection and classification on a global scale. In *European Conference on Computer Vision*, pages 68–84. Springer, 2020.

[2] D. Reis, J. Kupec, J. Hong, and A. Daoudi. Real-time flying object detection with yolov8. *arXiv preprint arXiv:2305.09972*, 2023.