Real-Time Road Sign Detection and Classification

1st Dumitru Mițca
dept. CTI
Technical University "Gheorge Asachi" of Iași
Iași, Romania
dumitru.mitca@student.tuiasi.ro

2nd Mihai Mihalache
dept. CTI
Technical University "Gheorge Asachi" of Iași
Iași, Romania
mihai.mihalache2@student.tuiasi.ro

3rd Otilia Zvorișteanu
dept. CTI
Technical University "Gheorge Asachi" of Iași
Iași, Romania
otilia.zvoristeanu2@academic.tuiasi.ro

4th Simona Caraiman
dept. CTI
Technical University "Gheorge Asachi" of Iași
Iași, Romania
simona.caraiman@academic.tuiasi.ro

***Abstract*—Road signs are a vital part of traffic infrastructure and their correct detection and classification is a vital task in autonomous driving applications. In this paper, we present a cursory study over state-of-the-art software solutions and related papers in the field, and then we present our initial implementation of such an application.**

***Index Terms*—computer vision, road sign detection, autonomous driving**

I. Introduction

Road sign detection and classification is a significant task in the computer vision space, that has been studied for at least a decade and a half. Applications in this domain have strict requirements, as mistakes or software bugs can lead to significant financial loss or even human death in the worst case, requirements such as:

- low, tending towards 0%, misclassification rate — classifying a stop sign as a right-of-way sign can lead to accidents, this example is unlikely to happen in a real application, but it shows the worst case scenario; more likely missclassication would be "must go left" as "must go right", or misclassifying the various "road thins ahead" signs between each other
- high speed — a vehicle should not stop at every road sign it encounters in order to recognize it then proceed, abiding its instructions, this process should be real-time
- high tolerance to bad weather conditions — fog, rain and snow are very common conditions in various parts of the world, an autonomous vehicle should be able to function correctly in any of these conditions. This point ties into the low missclassification rate point
- high tolerance to bad lightning — similar to the above
- low hardware cost — cars are already expensive, by including expensive high resolution cameras or expensive processors to run the detection and classification steps, the costs would be very unreasonable for the vast majority of people

Fig. 1. Three confusable signs, in order from left to right: "Îngustarea șoselei din ambele sensuri", "Îngustarea șoselei din dreapta", "Îngustarea șoselei din stânga"[2]

A. Related work

We've surveyed a number of papers in this domain in order to discover what approach we should follow in our own application.

Reference [1] was interesting to read but ultimately we found the approach in this paper to be limiting as it could only detect road signs with red borders that were either circles or triangles.

Reference [2] proposed an approach that had multiple layers of preprocessing before the detection and classification itself, which was done using the YOLO (You only look once) architecture.

Reference [3], a metastudy on the state of computer vision in traffic sign detection and recognition in 2019, was very useful as it showed us approaches that would be likely be useful with good accuracy and clear paths to take in development. It confirmed to us that an approach where the detection and recognition steps were split was a good idea as it is what most of the literature surveyed did. We initially believed that we could use grayscale datasets to simplify our machine learning (ML) models, but [3] infirmed this belief showing us that colour-based approaches would have higher accuracy.

[2]By Gigillo83, own work CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=40362644, https://commons.wikimedia.org/w/index.php?curid=40362647, https://commons.wikimedia.org/w/index.php?curid=40362645, stitched by Mițca Dumitru into a single image.

Reference [4] claims that a grayscale approach would be based on shape detection and be computationally expensive.

We approached [5] due to its generic title, but it is an application in traffic sign inventory management, and as such has less strict requirements than our application, and a much higher error rate is acceptable in its domain.

Reference [6] presents an approach where the detection of road signs is done using classical segmentation methods, while the recognition itself is done using a multilayer perceptron neural network.

Conclusions:

- this kind of application should be split in two big modules: a detection module and a recognition module, with any number of pre-processing before them
- the detection module can be implemented using classical methods, while the recognition module must be implemented using neural networks
- grayscale approaches are not computationally viable

B. State of the art

For existing software solutions we surveyed the pretrained YOLOv8 [7] model YOLOv8n and Roboflow.

1) YOLOv8: Through experimentation using the pretrained YOLOv8 model, we came to the conclusion that pretrained models are unlikely to be useful to particular applications, due to the pretrained models' generality and datasets where weather and or lightning conditions aren't varied enough. We believe using similar, or the same, architectures used by pretrained models or fine-tuning them is a good path forward however.

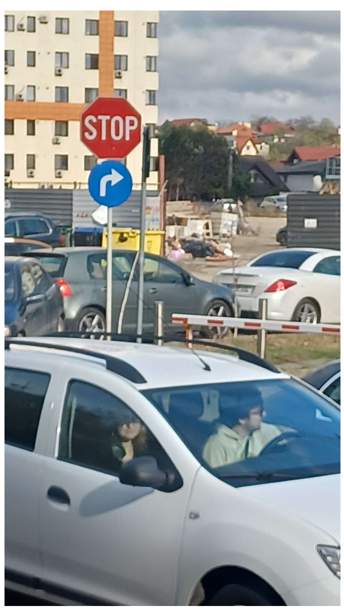

Fig. 2. YOLOv8n with no additional training detects 2 stop signs in this image

2) Roboflow: Roboflow[3] is a freemium service that provided us with two important services:

[3]Roboflow can be accessed at https://roboflow.com/

Fig. 3. Figure 2 with labels, as can be seen, the "must go right" sign is misclassified as a "stop sign"

- a dataset service that allows one to annotate datasets and also allows conversions between different dataset formats used by various preexisting architectures
- a model preview service, in which Roboflow will use your dataset to train a model that you can try in your browser — this service can be used a maximum of 3 times per project for free

However the freemium nature of the service is a disadvantage as one might invest a lot of time in this service due to its apparent free nature and latter be forced to buy-in when that might not have been the original intention.

We also noticed that there was considerable delay in using the models in one's browser, we are however not sure if this is a limitation of the models trained by Roboflow, the hardware on which they run, or the nature of the internet connection to it.

II. Method Description

We propose an approach in which the process is split in two steps: a detection step and a classification step, following existing practices in literature.

The detection step takes in an image, detects possible objects in the image that might be road-signs (based on color, shape, etc.). The classification step takes in an image containing only a possible road-sign and classifies it into the kind of road sign it is, or classifies it as a non-road-sign if the detection step misdetected an object as a road-sign. These steps are chained together using a middle layer that does any needed preprocessing on the segments from the original image given by the detection step, and then feeds them to the classification step.

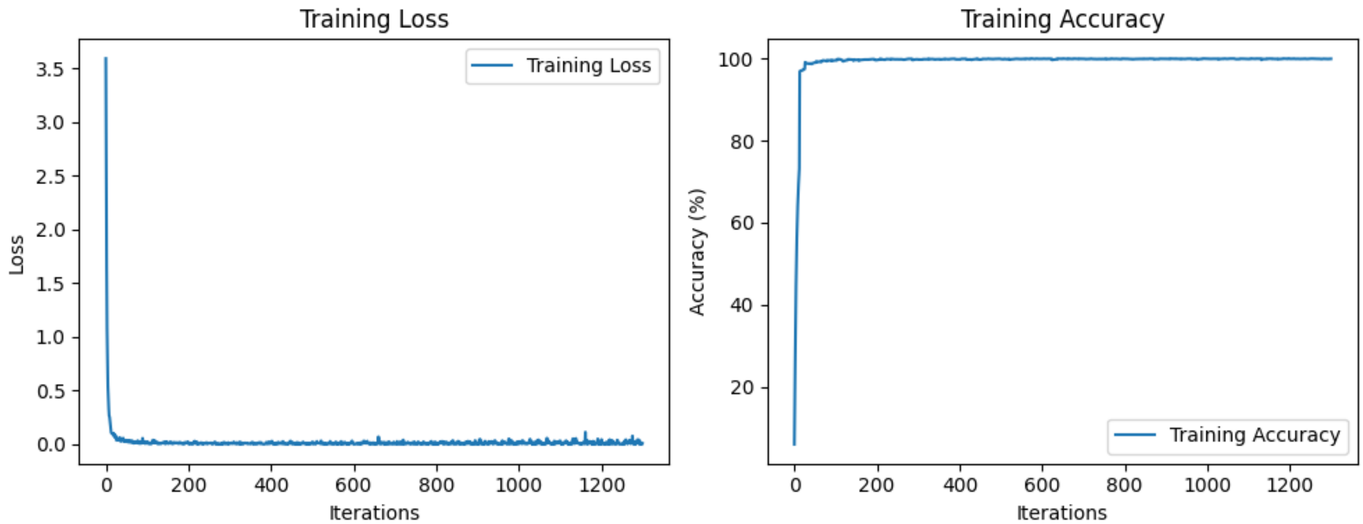

Fig. 4. Plot of the classification training progress showing a surprisingly rapid increase in accuracy

A. The detection step

We've chosen a neural-network-based approach for our detection step in order to see the what differences would arise from existing approaches taken in literature, but also in part due to perceived ease of development[4].

We've decided to use a model based on the YOLOv8 architecture that we've trained on various datasets, but have yet to find any success. Next we will explore a more classical approach to, hopefully, obtain more success in the detection step.

B. The classification step

We've chosen the German Traffic Sign Recognition Benchmark [8] for our dataset. The dataset resembled the CIFAR10 dataset so all we had to do is take an existing network that worked well on CIFAR10 and tweak it to work for our dataset. For training we used PyTorch [9] and got over 94% accuracy on the test set. Unfortunately we highly suspect that this method had overfit.

III. Preliminary results

As mentioned above we got over 94% accuracy on classification, and around 80% on segmentation but it perfomed badly on our personal test, that being showing a video of a car driving in town, or no detections on 2.

IV. Preliminary conclusions

Neural-network-based approaches are very accessible — *in theory* — but getting good results out of them is difficult due to not being aware of mistakes that can lead to overfitting or due to the powerful hardware needed for model training.

However, we do believe that real-time recognition of traffic signs becomes a more achievable goal as more and more time is passing due to this high accessibility, to people who are more experienced in the domain than us, and powerful hardware becoming more cheap with time.

[4]Reality often differs from one's expectations.

References

[1] R. Malik, J. Khurshid, and S. N. Ahmad, "Road sign detection and recognition using colour segmentation, shape analysis and template matching," in *2007 International Conference on Machine Learning and Cybernetics*, vol. 6, 2007, pp. 3556–3560.

[2] R. Karthika and L. Parameswaran, "A novel convolutional neural network based architecture for object detection and recognition with an application to traffic sign recognition from road scenes," *Pattern Recognition and Image Analysis*, vol. 32, no. 2, pp. 351–362, Jun 2022. [Online]. Available: https://doi.org/10.1134/S1054661822020110

[3] S. B. Wali, M. A. Abdullah, M. A. Hannan, A. Hussain, S. A. Samad, P. J. Ker, and M. B. Mansor, "Vision-based traffic sign detection and recognition systems: Current trends and challenges," *Sensors*, vol. 19, no. 9, p. 2093, May 2019. [Online]. Available: http://dx.doi.org/10.3390/s19092093

[4] A. Broggi, P. Cerri, P. Medici, P. P. Porta, and G. Ghisio, "Real time road signs recognition," in *2007 IEEE Intelligent Vehicles Symposium*, 2007, pp. 981–986.

[5] D. Tabernik and D. Skočaj, "Deep learning for large-scale traffic-sign detection and recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1427–1440, 2020.

[6] A. Hechri and A. Mtibaa, "Automatic detection and recognition of road sign for driver assistance system," in *2012 16th IEEE Mediterranean Electrotechnical Conference*, 2012, pp. 888–891.

[7] G. Jocher, A. Chaurasia, and J. Qiu, "YOLO by Ultralytics," Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[8] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark," in *International Joint Conference on Neural Networks*, no. 1288, 2013.

[9] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf