Traffic sign localization by triangulation through sequential detection and classification

Andrew Caunes[1, 3]
Loïc Alizon[1]
Thierry Chateau[1]
Philippe Balmas[1, 2]
Abir Gazzah[1, 2]
Vincent Frémont[3]

[1] Logiroad
[2] Université Clermont Auvergne
[3] Centrale Nantes, LS2N - Laboratoire des Sciences du Numérique de Nantes

andrew.caunes@logiroad.fr

Abstract

This article presents a novel approach to localize traffic signs on the road using computer vision techniques. Precise geo-localization and classification of traffic signs are crucial for applications such as autonomous driving, road infrastructure maintenance and traffic monitoring. Our proposed method leverages the power of machine and deep learning algorithms to tackle this complex task, which can be viewed as a combination of multiple interrelated subtasks in computer vision. We review the current state of the art in the field then provide a detailed explanation of our approach and of the experiments we made to try and improve it. We also release our own classification dataset for French traffic signs (FTSC) to facilitate future research in this field.

Résumé

Cet article présente une nouvelle méthode de localisation pour les panneaux de signalisation sur la route grâce aux techniques de vision par ordinateur. La géo-localisation précise et la classification des panneaux de signalisation sont cruciales pour des applications telles que la conduite autonome, la maintenance de l'infrastructure routière et l'analyse du trafic. Notre méthode tire parti de la puissance des algorithmes d'apprentissage automatique et d'apprentissage profond pour aborder cette tâche complexe, qui peut être considérée comme une combinaison de plusieurs sous-tâches de vision par ordinateur dépendant les unes des autres. Nous passons en revue l'état de l'art actuel dans le domaine et fournissons une explication détaillée de notre approche ainsi que des expériences menées pour l'améliorer. Nous publions également notre propre base de données de classification pour les panneaux de signalisation français (FTSC) pour faciliter les recherches futures dans ce domaine.

Keywords

Localization, Detection, Classification, Triangulation, Traffic Signs

1 Introduction

1.1 The task

This article deals with the task of accurate localization of road traffic signs. This task refers to the prediction of the geographic coordinates and class of each traffic sign in a given dataset. In our case, the dataset consists of natural images acquired from one or multiple 2D cameras mounted on a vehicle as well as the GPS positions of the camera(s) at the time of capture. There are variants of this setting where more features are provided in the dataset such as 3D point clouds obtained from a LiDAR sensor, but in this article we limit ourselves to the raw 2D images. The classes of the traffic signs are usually derived from a given catalog which may vary from a country to another. We explain the catalog we use in 2.3.

1.2 Applications

Localization of traffic signs on the road has been identified as a crucial aspect in the development of various applications in the field of transportation. Accurate and reliable detection and localization of traffic signs can aid in the development of autonomous driving systems as well as advanced driver assistance systems (ADAS) and other intelligent transportation systems (ITS) by providing them with important road information and guiding them to adhere to traffic regulations. Additionally, localization of traffic signs can assist in road maintenance and traffic monitoring by providing easy-to-update data on the state of the road infrastructure which allow prioritization of repairs and maintenance efforts. This information can also be used to improve traffic flow and reduce congestion by identifying bot-

tlenecks and potential hazards on the road.

1.3 Challenges

The task of predicting the location of objects in images captured by multiple cameras is a challenging problem, especially in the case of traffic signs and other road infrastructure. Gathering annotated data for this task is impractical due to the high cost and complexity involved. Fortunately, data can be obtained from sensors such as cameras, accelerometers or lidars mounted on vehicles a moderate cost, providing various features that can be used to analyze road infrastructure. Recent advances in deep learning technologies for computer vision have made it possible to accurately detect and classify objects in images and 3D point clouds in road datasets. These datasets often include information about the locations of the sensors at the time of capture. We leverage these datasets to predict the location of the objects using a method based on triangulation which we present in this article as a novel approach to traffic sign localization. The main challenges of this approach can be summarized as follows :

1. Compounded errors : The proposed pipeline is comprised of modules solving multiple interrelated sub-tasks, the final result is thus impacted by the compounded errors of the intermediary results. Therefore, each task must be solved accurately to ensure success on the final task.
2. Optimizing hyperparameters : Measuring performance on each individual task and optimizing hyperparameters based on validation scores does not guarantee an improvement of the final performance, as there may be exist a bias between the individual objectives and the final one.
3. Task design : There are multiple ways to design and solve individual tasks. For instance, traffic sign detections can be obtained either by applying sequentially one one-class detector and one multi-class classifier or by using a single multi-class detector. The first approach may be easier to implement whereas the latter may yield better results.
4. Domain adaptation : There may be a problem of domain shifting arising from using open datasets for learning some tasks and a different target dataset for the final task. The open datasets may not reflect the true distribution of the target dataset and this may cause the models to perform poorly in our final validation.

In this article, we present our method and show the results we obtained. We try to partially address some of these challenges, but more focus should be given to each of them to improve the performance further.

1.4 State of the art

The research on automated road infrastructure analysis, including traffic sign localization, is relatively thin compared to the interest it represents. However, recent advances in deep learning technologies and the availability of large open datasets have enabled significant progress in this field. The German Traffic Sign Recognition Benchmark (GTSRB) [21] and the German Traffic Sign Detection Benchmark (GTSDB) [12] for traffic sign classification and detection, respectively, have paved the way for large traffic sign datasets. Several other datasets have been released in recent years. The European Traffic Sign Dataset (ETSD), released with [20], combines multiple datasets from European countries, including GTSRB, and provides over 80,000 classification samples. This dataset offers a more comprehensive representation of European traffic signs and expands the scope of available training data for traffic sign analysis. Moreover, open datasets such as Mapillary Vistas [16] and Cityscapes [5] have also annotated traffic signs as a unique class for 2D object detection and semantic segmentation.

Segmentation, Detection and Classification have been the main focus of the community around automatic traffic sign analysis. Early works involved using image processing methods to recognize shapes and colors of traffic signs in natural images as well as to classify them [15]. Rapidly however, it became apparent that Traffic Sign Recognition (TSR) (or classification) was an ideal application for Deep Learning, since traffic signs can be differentiated given the adequate abstract features (shape, orientation, text, ...) but those features are hard to extract using shallow image processing methods. [3] applied a deep neural network on TSR before the deep learning revolution which came with [13], which itself mentions TSR as a traditional task solved by Convolutional Neural Networks (CNNs). Most state of the art work make use of recent deep learning technologies with variations. [11] presents an automatic recognition algorithm for traffic signs based on visual inspection. The authors propose a region of interest (ROI) extraction method through content analysis and key information recognition, a Histogram of Oriented Gradients (HOG) method for image detection, and a traffic sign recognition learning architecture based on CapsNet [18]. CapsNet relies on neurons to represent target parameters like dynamic routing, path pose and direction, and effectively captures traffic sign information from different angles or directions. More complex than simple TSR is the detection of traffic signs in natural images. Better performance should be obtained on the task of combined detection and classification when training a model to achieve both tasks simultaneously, but this is only possible provided datasets with appropriate labels. Some works provide models and training methods [1], [24] and some even provide additional data [27]. [22] presents an approach based on Mask R-CNN for detecting and recognizing a large number of traffic-sign categories suitable for automating traffic-sign inventory management. The authors propose several improvements to the Mask R-CNN architecture, including a region proposal network, anchor box alignment, and multi-scale testing. The propo-

sed approach is applied to the detection of 200 traffic-sign categories represented in a novel dataset, and results are reported on highly challenging traffic-sign categories that have not yet been considered in previous works. [26] addresses two main challenges in traffic sign detection : the difficulty of detecting small traffic signs and the interference caused by illumination variation and similar false traffic signs using a pyramid shape cascaded R-CNN as well as negative samples and specialized data augmentations.

Despite the progress made in detection and classification of traffic signs, little research has been made on road infrastructure geo-localization from street-level images and sensor location. The general idea of the state of the art methods is to leverage the previously mentioned solutions for detection and classification on street-view images and then use additional features such as locations of the sensors [17], street maps [25] or 3D lidar point clouds [7] to match the detections to locations. The method proposed in [25] involves segmenting objects in the images using convolutional neural networks, estimating their distance from the camera using monocular depth estimation, and geolocating them coherently using a custom Markov Random Field model for triangulation. [23] uses a modified version of RetinaNet (GPS-RetinaNet) to predict a positional offset for each sign relative to the camera, in addition to performing standard classification and bounding box regression. Candidate sign detections from GPS-RetinaNet are condensed into geolocalized signs by a custom tracker, which consists of a learned metric network and a variant of the Hungarian Algorithm.

Our method is based on a straightforward triangulation approach that utilizes detected and classified traffic signs from 2D camera data. Despite its simplicity, it has delivered promising results on our datasets and can be used as a benchmark for traffic sign localization tasks.

2 Method

This section outlines the traffic sign localization pipeline, which proceeds in a general-to-specific order. For each module, we provide details on the datasets and models employed, as well as the experiments conducted and validation results obtained.

2.1 Pipeline

To circumvent the lack of annotated data for direct traffic sign localization from 2D camera images and sensor coordinates, we split the task into multiple smaller, solvable sub-tasks and use the results from each task to achieve a satisfactory overall performance in localizing the objects. The global pipeline can be seen in Fig.1.

Our dataset is sourced from 2D cameras mounted on a vehicle, which capture geo-located images at 5-meter intervals. Our method utilizes this information to detect objects across multiple images, enabling us to triangulate their approximate location with a certain degree of uncertainty. Multiple conditions are required for the method to be applicable. While classical techniques such as convolutional neural networks (CNNs) can be used for object detection and classification, training is contingent on the availability of a sufficiently large dataset with annotations for the desired objects. Additionally, triangulation is reliant on accurate GPS locations and availability of camera characteristics such as internal and external orientation and distortion coefficients. Finally, to be located, an object must appear on multiple images from different cameras or the same camera in different locations.

Note that simultaneous detection and classification of objects in 2D images has been shown to improve performance compared to separate detection and classification methods, which may suffer from distribution gaps. However, this joint approach is hindered by the lack of labeled object detection data for all classes of traffic signs. While the French traffic sign catalog includes over 300 unique classes of signs, datasets for traffic sign detection often only provide samples for less than 200 classes, which may not align with a given application. On the other hand, there are large open road datasets that label traffic signs as a single class. These datasets typically consist of substantially more samples than specialized traffic sign datasets. For this reason, we choose the less optimal solution of training a detector first and a classifier second on individual datasets.

2.2 Detection

Datasets. There are many datasets of street-level images which annotate traffic signs either for object detection, semantic segmentation or both. Some specialized datasets distinguish between the traffic sign classes [12], [22], while some only label traffic signs as a unique class. We made the choice to split the detection and classification tasks because of the general larger size of the datasets annotating traffic signs as one class and in order to have better control over the catalog of signs we choose to classify.

We choose two common datasets to train our detector : Mapillary Vistas [16] and Cityscapes [5]. Mapillary Vistas is a vast dataset with over 25,000 high-resolution images covering a wide range of geographic locations and weather conditions, making it ideal for training models to detect traffic signs in diverse environments and domains. These open datasets offer a valuable resource for developing and testing traffic sign recognition algorithms and have facilitated rapid progress in this field in recent years. Similarly, Cityscapes is another popular dataset of street-level images among others. It consists of over 25,000 high-quality images with pixel-level annotations in different weather conditions and lighting situations. Characteristics for both dataset can be seen in Table 1.

We gathered our target dataset using multiple 2D RGB cameras mounted on a vehicle circulating around the city of Antony, France. The images are 2046x2046. Depending

on the camera, they may be partially occluded by parts of the vehicle. They were acquired over multiple days, during daytime, and under various weathers. While these factors can aid the model's ability to adapt to various domains, it would be ideal to include data from other domains, such as nighttime or heavy rain, to improve the model's robustness. To address this limitation, we tried multiple ways to mix the source datasets and studied the impact on the validation.

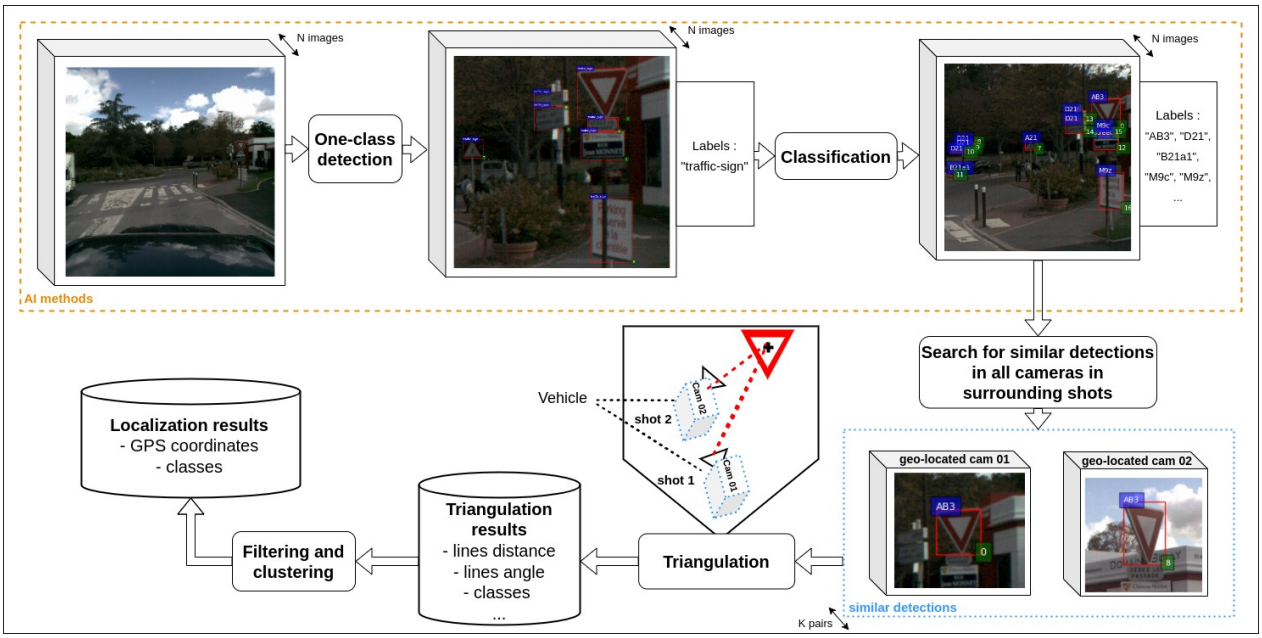

FIGURE 1 – Localization Pipeline.

Feature	Cityscapes	Mapillary Vistas
Number of coarse-grained samples	20,000	25,000
Number of fine-grained samples	5,000	0
Number of samples with TS labels	2,811	16,868
Image Resolution	2048 x 1024	various
Scene Diversity	Urban	Diverse
Annotated Classes	30	66
Countries of Origin	Germany	Worldwide

TABLE 1 – Characteristics of large Object Detection datasets which annotate traffic signs.

Model. Modern detectors are able to achieve impressive performance on various object detection tasks with relatively short training time. We choose to train multiple models using implementations from [2]. We pick the standard two-stage detector Faster-RCNN [19] for establishing a baseline but also train more advanced detectors such as TOOD [8] and Dynamic Head [6]. TOOD (Task-aligned One-stage Object Detection) is a one-stage object detector that aims to explicitly align the tasks of object classification and localization, which are usually implemented using heads with two parallel branches. TOOD proposes a novel task-aligned head that balances learning task-interactive and task-specific features, and a task aligment learning method to learn the alignment between the two tasks during training. Dynamic Head is a novel framework for unifying object detection heads with attention. It combines multiple self-attention mechanisms for scale-awareness, spatial-awareness, and task-awareness to improve the representation ability of object detection heads without any computational overhead.
For each model we choose to use a ResNet50 [10] backbone, we apply standard data augmentation from [2] including random resizing of the samples and random flips.

Experiments and validation. Our detection validation metric is the Average Precision (AP) metric computed in COCO format which we compute for the validation sets of both datasets. We then pick the best weights for each model according to the averaged AP to run further experiments with the rest of the pipeline.
As a first experiment, we train multiple models with the previously mentioned architecture and compare their performance on the standard validation sets from Mapillary Vistas and Cityscapes, using only the traffic sign annotations. As a second experiment, we try to mix both training datasets in various ways by varying the number of times that each dataset is repeated in each epoch as well as whether we use only fine-grained annotated samples

from Cityscapes or both fine-grained and coarse grained samples. An important point to note is that not all samples in the datasets show traffic signs, so the training sets of Mapillary Vistas and Cityscapes (fine-grained only) respectively contain 16,868 and 2,811 annotated samples. The configurations we use are the following :

— **N1M1** : One epoch is 1 time the training set of Mapillary Vistas and 1 time the training set of Cityscapes (fine-grained only)
— **N1M6** : One epoch is 1 time the training set of Mapillary Vistas and 6 times the training set of Cityscapes (fine-grained only)
— **N1M1extra** : One epoch is 1 time the training set of Mapillary Vistas and 1 time the training set of Cityscapes (fine-grained and coarse-grained)

We only use the Dynamic Head architecture for the experiments on dataset configuration.
We train each model for 12 epochs with pretrained weights for the COCO dataset [14] provided by [2]. We use a standard learning rate of 0.1 and scale it by a factor of 0.1 at epochs 8 and 11. We train on multiple machines with GPUs of 8 to 12 GiB of RAM. The batch sizes used vary between 1 and 4, we apply learning rate scaling according to [9] for optimized performance.

The validation results can be seen in Table 2 and 3. Our primary interest which is the impact on the final performance of the pipeline will be reported in the following sections.

Dataset Configuration	**mAP** (Cityscapes)	**mAP** (Mapillary)	**mAP (Avg)**
N1M1	0.437	0.342	0.3895
N1M6	**0.439**	**0.347**	**0.393**
N1M1extra	0.354	0.34	0.347

TABLE 2 – Experiment 1. Validation for traffic sign detection with a Dynamic Head architecture (DyHead) [6] trained with different configurations of the Mapillary Vistas [16] and Cityscapes [5] datasets.

Model	**mAP** (Cityscapes)	**mAP** (Mapillary)	**mAP (Avg)**
DyHead	0.439	0.347	0.393
TOOD	**0.452**	**0.355**	**0.4035**
Faster-rcnn	0.421	0.31	0.3655

TABLE 3 – Experiment 2. Validation for traffic sign detection with various architectures : Dynamic Head [6], TOOD [8] and Faster-RCNN [19].

According to this preliminary validation, the **N1M6** dataset configuration is superior to the others and the TOOD architecture surpasses DyHead and Faster-RCNN in terms of mAP.

2.3 Classification

Datasets. Several datasets have been developed for traffic sign classification over the years as seen in 1.4. They often vary regarding sample size and conditions of acquisition (weather, time of the day, etc.), and an important distinction resides in the traffic sign classes used for the labels. This is unique to traffic signs as there exists multiple catalogs providing different classes signs that do or do not look the same depending on the country of origin. Some applications require to be able to classify a large number of classes but most of the existing datasets only annotate up to 200 classes [22] while there are, for instance, more than 300 classes in the French catalog.

We release with this paper the French Traffic Sign Classification dataset (**FTSC**), which contains over 10,000 hand-annotated samples of traffic signs captured in the city of Antony, France, using our own equipment. The images were taken over multiple days with varying weather conditions, and their resolution ranges from around 15x15 to 800x800 pixels. To annotate the dataset, we used a catalog of 82 classes derived from the French traffic sign catalog, and we also included 9 'Not In catalog' (NIC) classes for frequently detected objects such as license plates and street plaques.

FIGURE 2 – Sample images from the FTSC dataset. The first row shows the intra-class variability. The second row shows various classes ('*B14_30*', '*D21*', '*B1*' and '*AB1*' respectively). Notice the variability in the angles and resolutions of the image. We hope that this characteristic will aid with models robustness to these changes. The last row shows some of the "Not-In-catalog" classes ('*others*', '*street*', '*license_plate*' and '*backwards_round*' respectively)

We index our annotations on the system of categories and superclasses used in the ETSD [20] which itself used a sys-

tem proposed with the GTSRB dataset [21]. This is a subdivision of the individual traffic sign classes into categories such as *Informative*, *Regulatory* and *Danger*. We further divide these categories into superclasses. For instance, the *Regulatory* category is divided between the superclasses *Priority*, *Mandatory*, *Prohibitory*, to which we also add counterparts such as *End Mandatory*. We provide CSV files and scripts for easy manipulations of the dataset including changing from superclasses format to simple sign labels or bare images as well as to convert the annotations from our proposed catalog to the catalog used by other datasets. This material as well as the dataset will be accessible online at https://github.com/andrewcaunes. Samples from the dataset can be seen in Figure 2.

Experiments and Validation. For our experiments, we trained multiple ResNet50 models using the ETSD dataset, the French Traffic Sign Classification dataset (FTSC), and a much smaller dataset that we refer to as SignDataset (SD). This last dataset is of lower quality but represents an interesting addition since it uses virtual traffic signs (see Figure 2.3) to increase the number of classes represented in the dataset. The hope with this addition is that with enough data transformation and from the knowledge of the nature of real traffic signs brought by the rest of the dataset, the model may be able to learn new traffic sign classes with little to no real samples.

((a)) A15c ((b)) B41 ((c)) M5 ((d)) CE5b

FIGURE 3 – Virtual signs included in SignDataset to increase the number of represented classes.

We implement our experiments using the mmclassification library [4]. In total, we used relatively few training samples from SD, as we focused primarily on ETSD and FTSC. We experimented with various configurations, including :

- FTSC (F) : The training set is from FTSC only.
- FTSC-ETSD (FE) : The training set is the concatenation of FTSC and ETSD.
- FTSC-ETSD-SD (FES) : The training set is the concatenation of FTSC, ETSD and SD.
- FTSC-noNIC (FN) : Same as (F) but we remove the 9 'Not-In-catalog' classes.
- FTSC-ETSD-noNIC (FEN) : Same as (FE) but we remove the 9 'Not-In-catalog' classes.
- FTSC-ETSD-SD-noNIC (FESN) : Same as (FES) but we remove the 9 'Not-In-catalog' classes.

Table 4 recapitulates the characteristics of the various configurations, including the number of training samples and the number of times we repeat the wrapped dataset (to make up for the inequalities of epoch length between configuration).

Dataset Configuration	**Training samples**	**N (repetitions)**	**K (classes)**
(F)	10,959	20	91
(FE)	67,273	4	160
(FES)	73,911	4	175
(FN)	9,116	20	82
(FEN)	65,430	4	166
(FESN)	72,054	4	151

TABLE 4 – Dataset configurations for experiment 3.

Dataset	**mAP**
(F)	99.1758
(FE)	98.5365
(FES)	94.9861
(FN)	**100.0**
(FEN)	98.6282
(FESN)	96.4131

TABLE 5 – Experiment 3. Validation for traffic sign classification depending on the dataset configuration.

The evaluation is made using the average precision metric on validation sets that are chosen are either 1% or 2% of the respective training datasets. Comparison between the experiments using this validation is only partially relevant since the validation set changes for each experiment. It is however not possible to pick a unique fair evaluation set since each dataset configuration annotate a different number of classes. We will be able to compare the results in a more relevant manner in the next section when we look at the performance of the global pipeline. The validation results can be seen in 5.

2.4 Triangulation

Algorithm. Given the detected and classified traffic signs in a dataset of natural images coming from regularly spaced shots from multiple cameras and knowing the characteristics of these cameras including their GPS locations, we are able to determine the angles of the traffic signs relatively to the cameras and therefore apply triangulation to locate the signs. The algorithm is described in 1.

Input : Street-level images and GPS coordinates of cameras
Output : Locations and classes of traffic signs
Detect and classify all traffic signs on all cameras and shots ;
foreach *shot* **do**
  foreach *detected sign* **do**
    Check in the surrounding shots if any camera has detections of the same class ;
    foreach *pair of similar detections* **do**
      Compute the angles of the detections in the horizontal plane with the known characteristics of the cameras ;
      Find a point that minimises the distance to the two lines ;
      Triangulate points and angles between lines ;
    end
  end
end
Apply clustering to find the plausible locations of the traffic signs ;
Return the locations and classes of the signs ;

Algorithme 1 : Traffic Sign Localization by Triangulation.

Experiments and validation. We conducted experiments with the pipeline using data collected in Antony, France, using seven RGB cameras to capture the surroundings of a vehicle. Modern smartphone devices can provide the same kind of data and could theoretically be use to obtain similar results.

To validate our method, we applied the pipeline on 5,603 shots, each containing seven images, in an area where the traffic signs were accurately geo-located. To compare our predictions to the ground-truth in terms of precision, recall, and F1-score, we created definitions for the predicted samples :

— A predicted sign is a true positive if it is within a minimum distance of d of a ground-truth sign of the same class.
— A predicted sign is a false positive if no ground-truth sign of the same class can be found within the minimum distance d.
— A ground-truth sign is a false negative if no predicted sign of the same class can be found within the minimum distance d.

Using these definitions, we computed precision, recall, and F1-score for all experiments on the detector and classifier. We alternatively ran the pipeline with a given classifier for experiments on the detector and vice versa. The final results for all experiments can be found in Table 6.

Our system achieves F1-scores of up to 85% for traffic sign localization at a minimum distance of 2 meters, which is sufficient for assisting experts in road infrastructure maintenance. However, there is potential for further improvement by optimizing the numerous hyperparameters involved in the pipeline. These include the parameters for training the detector and classifier, as well as those affecting the triangulation algorithm, such as filtration parameters applied on the confidences of detection and classification, triangulation angle, and distance of prediction. Additionally, our results may be impacted by our choice of traffic sign catalog, which may not align perfectly with the one used in the annotated data for the localization task. Therefore, the actual performance of our system may be higher than reported.

Tables 8 and 9 present the localization performance of experiments 2 and 3. The results indicate that the Dynamic Head architecture for the detector outperforms TOOD and Faster-RCNN by more than 5 points in F1-score for $d = 2m$ and more than 11 points for $d = 1m$. While Table 3 shows that TOOD has a better mAP on the validation set, it is important to note that the validation of each module in the pipeline can be challenging as it may not fully reflect its impact on the final performance of the entire pipeline.

Exp.	detector	classifier	P (d=2m)	R (d=2m)	F1 (d=2m)	P (d=1m)	R (d=1m)	F1 (d=1m)
0	DyHead-N1M1	(FE)	**0.862**	0.835	0.848	**0.752**	0.761	**0.756**
1	DyHead-N1M6	(FE)	0.849	0.849	0.849	0.729	0.77	0.749
2	DyHead-N1M1extra	(FE)	0.859	0.843	**0.85**	0.747	**0.762**	0.754
3	TOOD-N1M6	(FE)	0.735	0.871	0.798	0.56	0.73	0.634
4	Faster-RCNN-N1M6	(FE)	0.703	**0.9**	0.789	0.532	0.764	0.628
5	TOOD-N1M6	(F)	0.737	0.87	0.798	0.567	0.737	0.641
6	TOOD-N1M6	(FES)	0.748	0.867	0.803	0.574	0.723	0.64
7	TOOD-N1M6	(FN)	0.73	0.864	0.791	0.55	0.715	0.6225
8	TOOD-N1M6	(FEN)	0.745	0.872	0.803	0.561	0.721	0.631
9	TOOD-N1M6	(FESN)	0.726	0.864	0.789	0.552	0.719	0.624

TABLE 6 – Validation of the localization pipeline.

Dataset Config.	F1 (d=2m)	F1 (d=1m)
N1M1	0.848	**0.756**
N1M6	0.849	0.749
N1M1extra	**0.85**	0.754

TABLE 7 – Validation of Experiment 1 on the localization pipeline.

Detector	F1 (d=2m)	F1 (d=1m)
DyHead	**0.849**	**0.749**
TOOD	0.798	0.634
Faster-RCNN	0.789	0.628

TABLE 8 – Validation of Experiment 2 on the localization pipeline.

The validation for Experiment 1 based on the localization performance is presented in Table 7. The differences in performance are negligible, indicating that there is no significant impact when adding or removing the virtual samples from SD or the "Not-In-Catalogue" samples from the FTSC dataset.

classifier	F1 (d=2m)	F1 (d=1m)
ResNet50 (F)	0.798	0.641
ResNet50 (FE)	0.798	0.634
ResNet50 (FES)	**0.803**	**0.723**
ResNet50 (FN)	0.791	0.6225
ResNet50 (FEN)	**0.803**	0.631
ResNet50 (FESN)	0.789	0.624

TABLE 9 – Validation of Experiment 3 on the localization pipeline.

3 Discussion and Conclusion

In conclusion, we presented a novel approach to localize and classify traffic signs using computer vision techniques, which has demonstrated promising results. We leveraged advanced detector architectures and large object detection datasets to tackle this complex task, which is crucial for autonomous driving, road infrastructure maintenance, and traffic monitoring. We also provided a detailed explanation of our approach and the experiments we made to improve our proposed method. Moreover, we released our French traffic signs classification dataset to facilitate future research in this field.

Potential avenues for improvement resides in the more focused study of the challenges mentioned including simultaneous detection and classification, domain adaptation and careful design of the validation for the different modules of the pipeline. Another promising perspective would be to incorporate additional features such as 3D point clouds from lidars or geographic informations from open street maps in addition to the geo-located images. Future research could investigate the benefits of these potential improvements to advance the state of the art in traffic sign localization and classification.

Références

[1] J. CAO, C. SONG, S. PENG, F. XIAO, AND S. SONG, *Improved traffic sign detection and recognition algorithm for intelligent vehicles*, Sensors, 19 (2019).

[2] K. CHEN, J. WANG, J. PANG, Y. CAO, Y. XIONG, X. LI, S. SUN, W. FENG, Z. LIU, J. XU, Z. ZHANG, D. CHENG, C. ZHU, T. CHENG, Q. ZHAO, B. LI, X. LU, R. ZHU, Y. WU, J. DAI, J. WANG, J. SHI, W. OUYANG, C. C. LOY, AND D. LIN, *MMDetection : Open mmlab detection toolbox and benchmark*, arXiv preprint arXiv :1906.07155, (2019).

[3] D. CIREŞAN, U. MEIER, J. MASCI, AND J. SCHMIDHUBER, *Multi-column deep neural network for traffic sign classification*, Neural Networks, 32 (2012), pp. 333–338. Selected Papers from IJCNN 2011.

[4] M. CONTRIBUTORS, *Openmmlab's image classification toolbox and benchmark*. `https://github.com/open-mmlab/mmclassification`, 2020.

[5] M. CORDTS, M. OMRAN, S. RAMOS, T. REHFELD, M. ENZWEILER, R. BENENSON, U. FRANKE, S. ROTH, AND B. SCHIELE, *The cityscapes dataset for semantic urban scene understanding*, 2016.

[6] X. DAI, Y. CHEN, B. XIAO, D. CHEN, M. LIU, L. YUAN, AND L. ZHANG, *Dynamic head : Unifying object detection heads with attentions*, 2021.

[7] H. DOMÍNGUEZ, A. MORCILLO, M. SOILÁN, AND D. GONZÁLEZ-AGUILERA, *Automatic recognition and geolocation of vertical traffic signs based on artificial intelligence using a low-cost mapping mobile system*, Infrastructures, 7 (2022).

[8] C. FENG, Y. ZHONG, Y. GAO, M. R. SCOTT, AND W. HUANG, *Tood : Task-aligned one-stage object detection*, 2021.

[9] P. GOYAL, P. DOLLÁR, R. GIRSHICK, P. NOORDHUIS, L. WESOLOWSKI, A. KYROLA, A. TULLOCH, Y. JIA, AND K. HE, *Accurate, large minibatch sgd : Training imagenet in 1 hour*, 2017.

[10] K. HE, X. ZHANG, S. REN, AND J. SUN, *Deep residual learning for image recognition*, 2015.

[11] S. HE, L. CHEN, S. ZHANG, Z. GUO, P. SUN, H. LIU, AND H. LIU, *Automatic recognition of traffic signs based on visual inspection*, IEEE Access, 9 (2021), pp. 43253–43261.

[12] 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 The 2013 International Joint Conference on Neural Networks (IJCNN), 2013, pp. 1–8.

[13] Y. LECUN, Y. BENGIO, AND G. HINTON, *Deep learning*, Nature, 521 (2015), pp. 436–44.

[14] T.-Y. LIN, M. MAIRE, S. BELONGIE, L. BOURDEV, R. GIRSHICK, J. HAYS, P. PERONA, D. RAMANAN, C. L. ZITNICK, AND P. DOLLÁR, *Microsoft coco : Common objects in context*, 2014.

[15] A. MOGELMOSE, M. M. TRIVEDI, AND T. B. MOESLUND, *Vision-based traffic sign detection and analysis for intelligent driver assistance systems : Perspectives and survey*, IEEE Transactions on Intelligent Transportation Systems, 13 (2012), pp. 1484–1497.

[16] G. NEUHOLD, T. OLLMANN, S. R. BULÒ, AND P. KONTSCHIEDER, *The mapillary vistas dataset for semantic understanding of street scenes*, in 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 5000–5009.

[17] K. F. PEDERSEN AND K. TORP, *Geolocating traffic signs using large imagery datasets*, in 17th International Symposium on Spatial and Temporal Databases, SSTD '21, New York, NY, USA, 2021, Association for Computing Machinery, p. 34–43.

[18] K. QIAN, L. TIAN, Y. LIU, X. WEN, AND J. BAO, *Image robust recognition based on feature-entropy-oriented differential fusion capsule network*, Applied Intelligence, 51 (2021), p. 1108–1117.

[19] S. REN, K. HE, R. GIRSHICK, AND J. SUN, *Faster r-cnn : Towards real-time object detection with region proposal networks*, 2015.

[20] C. G. SERNA AND Y. RUICHEK, *Classification of traffic signs : The european dataset*, IEEE Access, 6 (2018), pp. 78136–78148.

[21] J. STALLKAMP, M. SCHLIPSING, J. SALMEN, AND C. IGEL, *Man vs. computer : Benchmarking machine learning algorithms for traffic sign recognition*, Neural Networks, 32 (2012), pp. 323–332. Selected Papers from IJCNN 2011.

[22] D. TABERNIK AND D. SKOČAJ, *Deep learning for large-scale traffic-sign detection and recognition*, IEEE Transactions on Intelligent Transportation Systems, 21 (2020), pp. 1427–1440.

[23] D. WILSON, T. ALSHAABI, C. VAN OORT, X. ZHANG, J. NELSON, AND S. WSHAH, *Object tracking and geo-localization from street images*, 2021.

[24] P. S. ZAKI, M. M. WILLIAM, B. K. SOLIMAN, K. G. ALEXSAN, K. KHALIL, AND M. EL-MOURSY, *Traffic signs detection and recognition system using deep learning*, 2020.

[25] C. ZHANG, H. FAN, W. LI, B. MAO, AND X. DING, *Automated detecting and placing road objects from street-level images*, 2019.

[26] J. ZHANG, Z. XIE, J. SUN, X. ZOU, AND J. WANG, *A cascaded r-cnn with multiscale attention and imbalanced samples for traffic sign detection*, IEEE Access, 8 (2020), pp. 29742–29754.

[27] Z. ZHU, D. LIANG, S. ZHANG, X. HUANG, B. LI, AND S. HU, *Traffic-sign detection and classification in the wild*, in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2110–2118.