Proceedings of Machine Learning Research 204:1–20, 2023 Conformal and Probabilistic Prediction with Applications

Confident Object Detection via Conformal Prediction and Conformal Risk Control: an Application to Railway Signaling

Léo Andéol LEO.ANDEOL@MATH.UNIV-TOULOUSE.FR
Institut de Mathématiques de Toulouse, Toulouse, France
SNCF, Saint-Denis, France

Thomas Fel THOMAS_FEL@BROWN.EDU
Brown University, Providence, Rhode Island, USA
SNCF, Saint-Denis, France

Florence de Grancey FLORENCE.DE-GRANCEY@IRT-SAINTEXUPERY.COM
Thales AVS France SAS

Luca Mossina LUCA.MOSSINA@IRT-SAINTEXUPERY.COM
IRT Saint Exupéry, Toulouse, France

Editor: Harris Papadopoulos, Khuong An Nguyen, Henrik Boström and Lars Carlsson

Abstract

Deploying deep learning models in real-world certified systems requires the ability to provide confidence estimates that accurately reflect their uncertainty. In this paper, we demonstrate the use of the conformal prediction framework to construct reliable and trustworthy predictors for detecting railway signals. Our approach is based on a novel dataset that includes images taken from the perspective of a train operator and state-of-the-art object detectors. We test several conformal approaches and introduce a new method based on conformal risk control. Our findings demonstrate the potential of the conformal prediction framework to evaluate model performance and provide practical guidance for achieving formally guaranteed uncertainty bounds.[1]

Keywords: Conformal Prediction, Object Detection, Uncertainty Quantification

1. Introduction

The deployment of Machine Learning (ML) technologies in real-world, safety-critical systems is faced with many challenges; one of them is to provide Uncertainty Quantification (UQ) for the output of the ML component. While this quantification can be accessible for low-complexity models, this is an important challenge for complex tasks such as object detection or text processing.

In this paper we explore how Conformal Prediction (Vovk et al., 2022, CP) and Conformal Risk Control (Angelopoulos et al., 2022, CRC) can contribute to build confident (or "trustworthy") predictors for the task of Object Detection (OD) (Zhao et al., 2019). CP and CRC have the advantage of being distribution-free, non-asymptotic and model-agnostic frameworks, which allow their deployment to any black-box predictor under minimal hypotheses, including complex ML tasks. Furthermore, they are computationally lightweight

[1] Our code is available at: https://github.com/leoandeol/conformal_railway_signal_detection

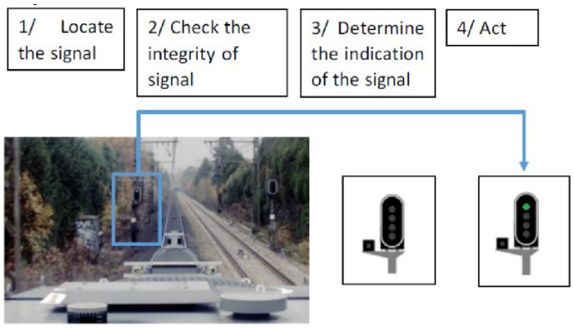

1/ Locate the signal
2/ Check the integrity of signal
3/ Determine the indication of the signal
4/ Act

Figure 1: Example of a pipeline where an Artificial Intelligence (AI) system acts following ML-based predictions. Source: Alecu et al. (2022).

as they do not require retraining the model, and so can easily be added to existing ML pipelines[2]. In this work, we demonstrate on a purpose-built dataset how the combination of these frameworks with state-of-the-art object detection models can lead to more accurate and reliable predictions in real-world applications. After introducing our use case in Section 1.1, we detail the construction of our railway signaling dataset in Section 2. Then, in Section 3 we provide an overview of CP, Conformal Risk Control (CRC) and some related methods. In Section 4 we give the details of our approach. Afterwards, in Section 5 we set up our experiments and discuss some important methodological details. Finally, in Section 5.3 and 6 we discuss the results, conclude on our work and give some insights, as well as leads for future works.

1.1. Railway Traffic Light Detection

Our use case consists in detecting the signals as they are encountered during the operation of trains in a railway network, and we refer to this problem as *Railway Traffic Light Detection* (RTLD). While main lines (e.g. high-speed lines) already have in-cabin signaling and can be automatized (Singh et al., 2021), this is too costly to be applied to the whole network. Consequently, on secondary lines, drivers can be subject to a larger cognitive load to interpret signals and the environment. Assisting drivers with AI-based signaling recognition could facilitate the operations. This operational desideratum can be cast as a functional chain including: locating the traffic light, validating the localization and recognizing the signal class. In Alecu et al. (2022), who provide an overview of the technical and regulatory challenges raised by the safety of AI systems in the railway and automotive industries, we find a study of this problem as depicted in Figure 1. Our application (referred to as *Traffic Light Localization*, TLL) would correspond to point (1).

1.2. Theoretical guarantees in ML with conformal prediction

Providing UQ for the output of an object detector can be interpreted as providing a set of points (e.g. pixels) that is likely to contain the (pixels of the) ground-truth box. Concretely, they can be computed by adding a "safety" margin around the sides of the predicted bounding boxes, according to some statistical property specified by the user. This could be the

[2] This holds true for *split* (or "inductive") Conformal Prediction (CP), which is the only form of CP we use for our applications.

average *coverage* of our procedure, that is, the frequency with which we capture the ground truth during inference. Letting X_{new} be the observed features of a new sample, we want a set predictor $\mathcal{C}(X_{\text{new}})$ that contains *entirely* the ground-truth box Y_{new} with probability:

$$\mathbb{P}\Big(Y_{\text{new}} \in \mathcal{C}(X_{\text{new}})\Big) \geq 1 - \alpha, \tag{1}$$

Equation 1 represents the core theoretical guarantee of the conformal prediction framework, which we will employ to analyze the outputs of our object detector. During inference, a conformal algorithm constructs a prediction set $\mathcal{C}_\alpha(X)$ that, on average, covers the observed value of the target Y with a frequency of $1-\alpha$ across multiple repetitions of the procedure. The challenge is to formulate the problem and build a set predictor $\mathcal{C}(\cdot)$ that answers to an operational need, such as "capturing the entire bounding boxes $(1-\alpha)$ 100% of the time" (for CP) or "covering at least $(1-\alpha)$ 100% of the target pixels" (for CRC, Section 4).

2. Building an experimental dataset for object detection

We work on the detection of traffic lights in the French railway network. Since most national railway networks have a unique mix of signals and traffic lights, one needs to build a dataset for the specific operational domain.

2.1. Related work

Public datasets for computer vision on railway data are rather scarce. Zendel et al. (2019) built the first public, railway-specific dataset for semantic scene understanding, which includes the manual annotation of geometric shapes and pixel-wise labeling. They also provide an overview of publicly available datasets containing in part railway data. To counter this scarcity, Gasparini et al. (2020) collected 30k night-time images, captured by a drone flying over the rails, for the task of detecting autonomously anomalous objects on rails. Another viable option is to use artificial data, as done for instance by Mauri et al. (2022), who created a virtual dataset from a simulator based on a video game. For the French network, Zouaoui et al. (2022) created a segmentation dataset with artificially generated anomalies. The dataset *FRSign* of Harb et al. (2020) addresses a similar use case via object detection: they were able to coordinate the data collection with the national railway operator and other partners.

In our case, we found an alternative sourcing existing videos from the internet. Also, compared to the work of Harb et al., our new dataset presents increased variability (more railway lines, environmental and weather conditions, etc.) which could enable more accurate predictions in real-world scenarios. Finally, we generalize the task from single to multi-object detection, laying the foundations for future work in instance segmentation.

2.2. Dataset characteristics

The Traffic Light Localization (TLL) should operate whenever a human operator typically operates. This leads to considering a high-diversity dataset, including various meteorological situations (rain, snow, etc.), various hours (night, day) and a wide variety of traffic light situations (fully observable, partially occluded by foliage, etc.). To partially account for these common issues, we included different light conditions in our data.

Table 1: Characteristics of our dataset

Characteristics	Quantity
Railway lines	41
Images per line (average)	83.27 ± 41.11
Images in dataset	3414
Dimensions (pixels)	1280 × 720
Bounding boxes per image	1.03 ± 1.26
Bounding boxes (total)	3508

As source data, we used footage from 41 videos of French railway lines available on the internet, with the approval of the uploader[3]. Most of the railway lines are distinct, but a few share sections especially around large Parisian stations. The average duration of a video is about 1.5 hours, from which we extracted individual frames: we run a pretrained object detector with a low objectness threshold, and we kept a minimum interval of 5 seconds between detections to avoid repeating the same signals and to prevent excessive temporal correlation between images. On average, 83 frames per video were extracted and manually annotated all visible railway traffic lights; a summary is given in Table 1.

3. Uncertainty quantification in object detection

For our tests, we restricted our attention to YOLOv5m (Jocher, 2020), originally proposed by (Redmon et al., 2016), DETR-ResNet50 (Carion et al., 2020) and DiffusionDet (Chen et al., 2022). YOLOv5m offers a one-stage detection, combining convolutional layers with regression and classification tasks, and has found widespread adoption. DETR-ResNet50 leverages transformer layers and DiffusionDet formulated OD as a denoising diffusion problem. These were chosen because they are either standard models, or state-of-the-art ones. Since CP and CRC are model-agnostic, the choice of OD network does not matter. For instance, Petrović et al. (2022) build a detector of railway tracks and signals. Also, the application of CP is not limited to traffic lights (our use case) but can be extended to any detection needing formal guarantees (Ye et al., 2020). Applying CP to specialized models could open up future lines of research.

3.1. Related works

In industrial applications, it is often hard to make reliable hypotheses on the data or the correct specification of the predictor, which is why we favored a distribution-free, model-agnostic approach such as conformal prediction. Of course, if one can make meaningful assumptions on their learning task, then there is a sizeable literature on the topic (Feng et al., 2021, for a review). Hall et al. (2020) give a probabilistic formulation of OD, where the probability distributions of the bounding boxes and classes are predicted. Bayesian models like in Harakeh et al. (2020) and Bayesian approximations (Deepshikha et al., 2021) are also found in the literature. We point out the distribution-free approach of Li et al. (2022): they build probably approximately correct prediction sets using a held-out calibration set

[3] We would like to thank the author of the Youtube channel: https://www.youtube.com/@mika67407

to compute a calibrated threshold for the predictor (e.g. for the softmax), following the principles introduced by Park et al. (2020). They control the coordinates of the boxes but also the proposal and objectness scores, resulting in more and larger boxes. Their method relies on the structure of Fast R-CNN (Ren et al., 2017), the underlying OD model: this has three detection steps with three predictors associated with the *proposal*, *presence* and *location* of a bounding box. Each component is controlled individually and then combined to attain the desired guarantee. Their method is an application of the PAC-based calibration of Park et al. (2020). This is not applicable *as is* to state-of-the-art one-stage object detectors such as *YOLO* (Redmon et al., 2016) or *DETR* (Carion et al., 2020). This is one of the reasons why we opt to model our uncertainty quantification problem directly via CP. Also, CP requires exchangeable data while concentration-based methods such as Park et al. (2020) and the more general methods of Bates et al. (2021) and Angelopoulos et al. (2021) require the stronger assumption of data being independently and identically distributed (i.i.d).

3.2. Principles of Conformal Prediction

Conformal Prediction (CP) (Vovk et al., 2022; Angelopoulos and Bates, 2023) is a family of methods to perform UQ with guarantees under the sole hypothesis of data being independent and identically distributed (or more generally exchangeable). CP is flexible because we can either "conformalize"[4] a predictor using the training data, for instance via transductive "full" CP (Vovk et al., 2022) or a K-fold partition scheme (Vovk, 2013; Barber et al., 2021), or via a dedicated calibration dataset D_{cal} with a method known as **split** CP (Papadopoulos et al., 2002; Lei et al., 2018). This allows using a **pretrained** predictor $\widehat{f}$ with no need to access the training data. It also requires only prediction set computation at inference, minimizing computational resources demands. This is a valuable aspect for embedded system, as is in TLL. So, throughout the paper, "CP" always refers to Split CP; unless otherwise specified, we write $n = |D_{\text{cal}}| = n_{\text{cal}}$ and (X_{n+1}, Y_{n+1}) will denote a (random) test point drawn from the same distribution as $D_{\text{cal}} = \{(X_i, Y_i)\}_{i-1}^{n}, (X_i, Y_i) \sim \mathbb{P}_{XY}$.

For a specified (small) error rate $\alpha \in (0,1)$ and n calibration points, during inference, the CP procedure will yield a prediction set $C_\alpha(X_{n+1})$ that fails to cover the observed Y_{n+1} with probability at most α:

$$\mathbb{P}\Big(Y_{n+1} \notin \mathcal{C}_\alpha(X_{n+1})\Big) \leq \alpha. \tag{2}$$

Formally, this guarantee holds true, on average, over many repetitions of the CP procedure (i.e. sampling of calibration and test points). It is valid for any distribution $\mathbb{P}_{XY}$, any sample size and any predictive model $\widehat{f}$, even if it is misspecified or a black box. The probability $1-\alpha$ is referred to as the *nominal coverage*; the *empirical coverage* on n_{test} test points is $\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \mathbb{1}\{Y_i \in C_\alpha(X_i)\}$.

The conformalization of $\widehat{f}$ is determined by a **nonconformity score** $s(X, Y)$, measuring how "unusual" the prediction $\widehat{Y} = \widehat{f}(X)$ is with respect to observed Y. This is a generic method: e.g., for regression we can set $s(X, Y) = |\widehat{f}(X) - Y|$; for quantile regres-

[4] We (loosely) say that we "conformalize" a predictor $\widehat{f}$ whenever we apply either CP or CRC. For CP, a *conformalized* model is one that outputs a prediction set (e.g. enlarged bounding box) that does not contain the target Y at most with frequency α.

sion (Koenker and Bassett, 1978), we can measure the errors of lower and upper quantile estimators $(\widehat{q}_\beta, \widehat{q}_{1-\beta})$ with $s(X,Y) = \max\{\widehat{q}_\beta(X) - Y, Y - \widehat{q}_{1-\beta}(X)\}$ of Romano et al. (2019).

```
Algorithm 1: Split conformal prediction: fit, conformalization and inference steps.
Input: Training data $D_{\text{train}} = \{(X_i, Y_i)\}_{i=1}^{n_{\text{train}}}$; miscoverage level $\alpha \in (0,1)$;
       nonconformity score $s(X,Y)$.
  1. Split (disjointly) training data: $D_{\text{train}} = D_{\text{fit}} \uplus D_{\text{cal}}$
  2. Fit (or fine-tune) $\widehat{f}(\cdot)$ on $D_{\text{fit}}$
  3. Compute scores on $D_{\text{cal}}$: $\bar{R} = \{s(X_i, Y_i)\}_{i=1}^{n_{\text{cal}}}$
  4. Compute conformal quantile: $q_{1-\alpha} = \lceil (n_{\text{cal}} + 1)(1-\alpha) \rceil$-th element of the sorted
     sequence $\bar{R}$
  5. Inference: $\mathcal{C}_\alpha(X_{n+1}) = \big\{ y : s(X_{n+1}, y) \leq q_{1-\alpha} \big\}$.
```

In Algorithm 1 we give the steps of Split CP. If one uses a pretrained predictor, then only D_{cal} are needed and Steps 1 and 2 are skipped. During conformalization, we compute the nonconformity scores $\bar{R}$ on D_{cal}. For a test point X_{n+1}, we build the prediction set $\mathcal{C}_\alpha(X_{n+1}) = \{y : s(X_{n+1}, y) \leq q_{1-\alpha}\}$. For example, if $s(X,Y) = |Y - \widehat{f}(X)|$, then we build the prediction interval as $\mathcal{C}_\alpha(X_{n+1}) = [\widehat{Y} - q_{1-\alpha}, \widehat{Y} + q_{1-\alpha}]$. In Section 4 we show that, for our TLL case, CP boils down to adding a margin around the predicted bounding boxes.

3.3. Conformal risk control: a generalization of conformal prediction

Angelopoulos et al. (2022) introduced Conformal Risk Control (CRC) as an extension of CP. First, they point out that the conformal guarantee in Equation 2 can be rewritten as $\mathbb{E}[\mathbb{1}\{Y_{\text{new}} \notin \mathcal{C}_\alpha(X_{n+1})\}] \leq \alpha$. The function $\ell\big(\mathcal{C}_\alpha(X_{n+1}), Y_{\text{new}}\big) = \mathbb{1}\{Y_{\text{new}} \notin \mathcal{C}_\alpha(X_{n+1})\}$ encapsulates a *notion of error*, which for CP occurs whenever Y_{new} is not covered by $\mathcal{C}_\alpha(X)$. In some practical applications, this binary loss can be too strict and building a prediction set according to another criterion can satisfy (theoretically) a different operational need (e.g. false negative rate). The CP procedure can be extended to any bounded loss function $\ell(\cdot)$, provided that it decreases as the set $\mathcal{C}(X_{n+1})$ gets larger; the task is generalized as $\mathbb{E}\big[\ell\big(\mathcal{C}(X_{n+1}), Y_{n+1}\big)\big] \leq \alpha$, where $\mathcal{C}(X_{n+1})$ is not necessarily built by CP. Let $\widehat{f}(X)$ be a pretrained predictor and $\mathcal{C}_\lambda(X; \widehat{f})$ a function parametrized by λ, where larger λ values yield larger prediction sets. Given a calibration dataset $D_{\text{cal}} = (X_i, Y_i)_{i=1}^n$, CRC boils down to computing the losses $L_i(\lambda) = \ell\big(\mathcal{C}_\lambda(X_i), Y_i\big) \in (-\infty, B], B < \infty$ on D_{cal}, the empirical risk $\widehat{R}_n(\lambda) = \frac{1}{n}(L_1(\lambda) + \cdots + L_n(\lambda))$ and choosing a $\widehat{\lambda}$ such that the risk on the $(n+1)$-th (unseen) sample is controlled:

$$\mathbb{E}\Big[\ell\big(\mathcal{C}_{\widehat{\lambda}}(X_{n+1}), Y_{n+1}\big)\Big] \leq \alpha. \tag{3}$$

For an arbitrary risk level upper bound $\alpha \in (-\infty, B]$, $\widehat{\lambda}$ is computed as:

$$\widehat{\lambda} := \inf\Big\{\lambda : \frac{n}{n+1}\widehat{R}_n(\lambda) + \frac{B}{n+1} \leq \alpha\Big\}. \tag{4}$$

Here, $\mathcal{C}_{\widehat{\lambda}}$ denotes any set predictor that complies with an α risk level, not necessarily a probability. Throughout the paper, however, we have losses $L(\lambda) \in [0,1]$ and $\alpha \in (0,1)$.

3.3.1. CRC covers the conformal prediction case

If we consider a miscoverage loss $L_i^{\text{coverage}}(\lambda) = \mathbb{1}\{Y_i \notin \widehat{\mathcal{C}}_\lambda(X_i)\} = \mathbb{1}\{s(X_i, Y_i) > \lambda\}$, Angelopoulos et al. (2022) shows that CRC finds the same prediction set as CP, for a given α. We can write the CRC prediction set as:

$$\mathcal{C}_{\hat{\lambda}}(X_{n+1}) = \{y : s(X_{n+1}, y) \leq \hat{\lambda}\}, \tag{5}$$

where $\hat{\lambda}$ is the same as the conformal quantile $q_{1-\alpha}$ of Line 5 of Algorithm 1 for Split CP. The CRC guarantee is less tight than the one proved by Lei et al. (2018) for Split CP, the latter being $1 - \alpha \leq \mathbb{P}(Y \in C_\alpha(X)) \leq 1 - \alpha + \frac{1}{n+1}$ while the former is $1 - \alpha \leq \mathbb{P}(Y \in C_{\hat{\lambda}}(X)) \leq 1 - \alpha + \frac{2B}{n+1}$.

4. Building Conformal Predictors for Object Detection

In our experiments, we test Conformal Prediction (CP) and Conformal Risk Control (CRC) in OD. For the first part, we follow the box-wise CP methods of de Grancey et al. (2022) and Andéol et al. (2023). To the best of our knowledge, these are the only straightforward applications of CP to object detection. In addition to their methods, we also test the new, better-performing, max-additive and max-multiplicative scores (see Section 4.1.1). For the second part of our experiments, we compare the image-wise CP method of de Grancey et al. (2022), to our approach which relies on the Conformal Risk Control (CRC) of Angelopoulos et al. (2022), which extends CP to a more general class of errors: while CP provides a guarantee on a binary error "the truth is *contained* vs *not contained* in the prediction set", CRC admits more generic guarantees of the type "$(1-\alpha)$ 100% of the pixels will be covered by the CRC output" (see Section 4.2).

We define objects we work with as follows: the output of the OD predictor is a variable-sized (potentially empty) set of bounding boxes $\widehat{f}(X_{n+1}) = \widehat{Y}_{n+1} = \{\widehat{Y}_{n+1}^k\}_{k=1,\ldots,n_i}$. However, unlike previous work of de Grancey et al. (2022) which considers conjunctions of half-spaces, we will note our bounding boxes in the common OD standard, as a set of four coordinates $\{\widehat{x}_{\text{min}}^k, \widehat{y}_{\text{min}}^k, \widehat{x}_{\text{max}}^k, \widehat{y}_{\text{max}}^k\}$ to ensure wider understanding among the OD community. However, this notation is imprecise, and therefore we also adopt an implicit definition of bounding boxes as the set of pixels that belong to them, which is closely related to segmentation and necessary for the proper definition of some of our proposed methods. Each box $\widehat{Y}_i^k$ is therefore the set of pixels

$$\widehat{Y}_i^k = \left\{ (x, y) \in \mathbb{R}^2 : \begin{array}{l} x \in [\ \widehat{x}_{\text{min}}^k\ ,\ \widehat{x}_{\text{max}}^k] \\ y \in [\ \widehat{y}_{\text{min}}^k\ ,\ \widehat{y}_{\text{max}}^k\ ] \end{array} \right\}.$$

In all cases, ground-truth boxes are defined equivalently. We can now introduce two different approaches to guaranteeing OD predictions, box-wise and image-wise guarantees.

4.1. Box-wise Conformalization

The most intuitive approach to conformalize object detector predictions is to work box-wise, that is to consider our Y_i as individual boxes, and compute residuals, as well as obtain guarantees in expectation, for individual boxes. However, this approach presents a challenge in defining the nonconformity score because the model only provides a set of predicted boxes. To address this, a pairing between predicted and ground-truth boxes is necessary, which is commonly done using the Hungarian matching algorithm in the object detection literature. Since the nonconformity scores, and consequently the conformal guarantee, depend on this pairing, it is regarded as an integral part of the conformalization procedure. This operation is typically performed using a specific criterion.

We consider a threshold on the IoU (Intersection over Union) score of boxes: a prediction may be considered matched if it has a sufficient score. The spectrum of predictions the guarantee applies to, as well as the size of margins depends strongly on this threshold. It is important to note that this guarantee will apply exclusively to true positives.

We further define multiple nonconformity scores, per coordinate, or a unique one per box, be it additive or multiplicative (as seen on Fig. 2). We follow de Grancey et al. (2022) and the Split CP presented in Algorithm 1 with the addition of the aforementioned matching rule.

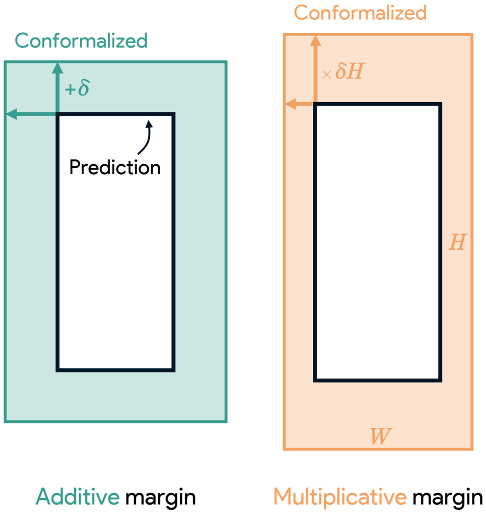


Figure 2: **Effect of margin systems used.** An additive margin is a number of pixels to add, while a multiplicative margin is a proportion of the width/height to add. The additive one may lead to comparatively a smaller effect on foreground boxes and larger on background (smaller traffic signals) boxes, and the opposite applies for multiplicative margins.

4.1.1. Nonconformity scores for object detection

Let $k = 1, \ldots, n_{box}$ index every ground-truth box in D_{cal} that was detected by $\widehat{f}$, disregarding their source image. Let $Y^k = (x^k_{\text{min}}, y^k_{\text{min}}, x^k_{\text{max}}, y^k_{\text{max}})$ be the coordinates of the k-th box and $\widehat{Y}^k = (\hat{x}^k_{\text{min}}, \hat{y}^k_{\text{min}}, \hat{x}^k_{\text{max}}, \hat{y}^k_{\text{max}})$ its prediction.

The nonconformity score, which we refer to as **additive**, is defined as:

$$R_k = \left(\hat{x}^k_{\text{min}} - x^k_{\text{min}},\, \hat{y}^k_{\text{min}} - y^k_{\text{min}},\, x^k_{\text{max}} - \hat{x}^k_{\text{max}},\, y^k_{\text{max}} - \hat{y}^k_{\text{max}}\right) \quad \text{(additive score).} \tag{6}$$

Also hinted by de Grancey et al. (2022), a **multiplicative** score can be defined:

$$R_k = \left(\frac{\hat{x}^k_{\text{min}} - x^k_{\text{min}}}{\widehat{w}^k}, \frac{\hat{y}^k_{\text{min}} - y^k_{\text{min}}}{\widehat{h}^k}, \frac{x^k_{\text{max}} - \hat{x}^k_{\text{max}}}{\widehat{w}^k}, \frac{y^k_{\text{max}} - \hat{y}^k_{\text{max}}}{\widehat{h}^k}\right) \quad \text{(multiplicative score),} \tag{7}$$

where the prediction errors are scaled by the predicted width $\widehat{w}^k$ and height $\widehat{h}^k$. Since they aim to capture a bounding box, which is represented by four coordinates, this yields a multidimensional response. This case is similar to multiple hypothesis testing, and to guarantee the coverage of the whole box, that is, the four coordinates at the same time, it is necessary to apply a statistical adjustment to the risk level α; and they opt for a Bonferroni correction where, for each coordinate, the prediction set is built with a miscoverage rate of $\frac{\alpha}{4}$ (see Equation 12 & 13).

It is well-known (Bland and Altman, 1995) that the Bonferroni correction can be overly conservative. To counter this, we propose and test the **max-additive** and **max-multiplicative** nonconformity scores, which are defined respectively as:

$$R_k^{\max} = \max\{\hat{x}^k_{\min} - x^k_{\min},\, \hat{y}^k_{\min} - y^k_{\min},\, x^k_{\max} - \hat{x}^k_{\max},\, y^k_{\max} - \hat{y}^k_{\max}\} \quad \text{(max-additive)}, \tag{8}$$

$$R_k^{\max} = \max\{\frac{\hat{x}^k_{\min} - x^k_{\min}}{\widehat{w}^k}, \frac{\hat{y}^k_{\min} - y^k_{\min}}{\widehat{h}^k}, \frac{x^k_{\max} - \hat{x}^k_{\max}}{\widehat{w}^k}, \frac{y^k_{\max} - \hat{y}^k_{\max}}{\widehat{h}^k}\} \quad \text{(max-multip.)}. \tag{9}$$

Both approaches are explained and compared on Figure 3. In general, it is not possible to determine a priori whether the additive or multiplicative score should be used. For instance, de Grancey et al. (2022) have operational reasons to prefer the additive margin: they are detecting pedestrians from the point of view of vehicles; when objects are close to the camera, they have larger bounding boxes and multiplicative conformalization would yield margins that are too large to be operationally useful.

4.1.2. Computing the conformal quantile

After the nonconformity scores, the next quantity to compute is the conformal quantile. For the case of additive and multiplicative scores with Bonferroni correction, we do:

$$q^c_{1-\frac{\alpha}{4}} = \lceil (n_{box}+1)(1-\frac{\alpha}{4})\rceil\text{-th element of the sorted } \bar{R}^c, \forall c \in \{x_{\min}, y_{\min}, x_{\max}, y_{\max}\}. \tag{10}$$

For the case of max-additive and max-multiplicative scores, the quantile is given by:

$$q_{1-\alpha} = \lceil (n_{box}+1)(1-\alpha)\rceil\text{-th element of the sorted } \bar{R}^{\max} \tag{11}$$

4.1.3. Computing the prediction set

During inference, for a new observation X_{n+1}, the **coordinates** of the additive and the multiplicative split conformal prediction boxes are given by:

$$\widehat{C}_\alpha(X_{n+1}) = \{\widehat{x}_{\min} - q^{x_{\min}}_{1-\frac{\alpha}{4}},\, \widehat{y}_{\min} - q^{y_{\min}}_{1-\frac{\alpha}{4}},\, \widehat{x}_{\max} + q^{x_{\max}}_{1-\frac{\alpha}{4}},\, \widehat{y}_{\max} + q^{y_{\max}}_{1-\frac{\alpha}{4}}\}, \quad \text{(additive)} \tag{12}$$

$$\begin{aligned}\widehat{C}_\alpha(X_{n+1}) = \{&\widehat{x}_{\min} - \widehat{w}\cdot q^{x_{\min}}_{1-\frac{\alpha}{4}},\, \widehat{y}_{\min} - \widehat{h}\cdot q^{y_{\min}}_{1-\frac{\alpha}{4}},\\ &\widehat{x}_{\max} + \widehat{w}\cdot q^{x_{\max}}_{1-\frac{\alpha}{4}},\, \widehat{y}_{\max} + \widehat{h}\cdot q^{y_{\max}}_{1-\frac{\alpha}{4}}\}. \quad \text{(multiplicative)}\end{aligned} \tag{13}$$

For the **max-error** conformalized versions, we get respectively:

$$\begin{aligned}\widehat{C}_\alpha(X_{n+1}) = \Big\{&\widehat{x}_{\min} - q_{1-\alpha},\, \widehat{y}_{\min} - q_{1-\alpha},\\ &\widehat{x}_{\max} + q_{1-\alpha},\, \widehat{y}_{\max} + q_{1-\alpha}\Big\}. \quad \text{(max-additive)}\end{aligned} \tag{14}$$

$$\begin{aligned}\widehat{C}_\alpha(X_{n+1}) = \Big\{&\widehat{x}_{\min} - \widehat{w}\cdot q_{1-\alpha},\, \widehat{y}_{\min} - \widehat{h}\cdot q_{1-\alpha},\\ &\widehat{x}_{\max} + \widehat{w}\cdot q_{1-\alpha},\, \widehat{y}_{\max} + \widehat{h}\cdot q_{1-\alpha}\Big\}. \quad \text{(max-multiplicative box)}\end{aligned} \tag{15}$$

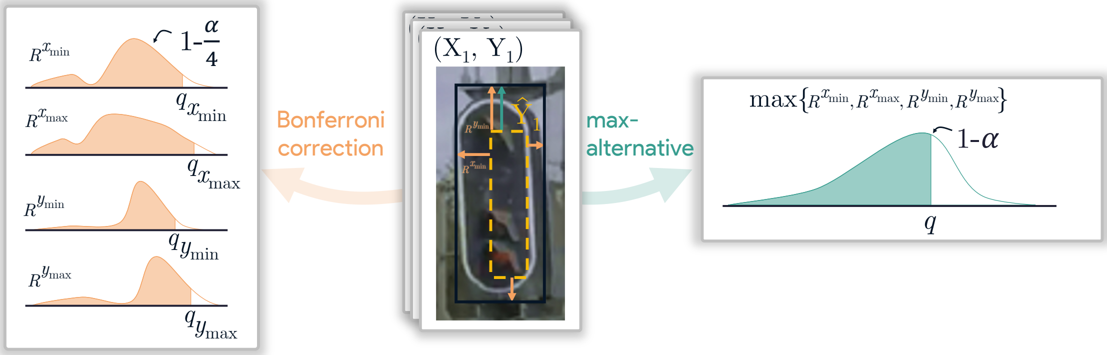


Figure 3: **Comparison of the previous Bonferroni approach vs our max-nonconformity score.** As explained in Section 4.1.1, we propose an alternative to the previously used Bonferroni correction which is overly conservative - as illustrated on the left, where four quantiles at the level $\frac{\alpha}{4}$ need to be estimated – one per coordinate of the bounding box. The alternative - illustrated on the right - consists in calculating the quantile at the level α of the distribution of coordinate-wise maximum residuals.

4.2. Image-wise Conformalization and Conformal Risk Control

Image-wise conformalization, as opposed to box-wise, considers our ground truth Y_i to be defined as a set of bounding boxes. We therefore measure a nonconformity score at the image level and do not need a pairing algorithm. However, here we focus on avoiding false negatives, and do not consider the false positive rate in any of the following methods, and it does not affect the process of conformal prediction or risk control.

We compare here two families of approaches. The first is inherited from de Grancey et al. (2022) and is rooted in a signed asymmetric Hausdorff distance. This method is therefore referred to as "Hausdorff"our experiments. The nonconformity score for this method is defined as the smallest margin such that, for a given image, a proportion $1 - \beta$ of ground-truth boxes is *entirely* covered by prediction boxes (even if it takes two prediction boxes that each cover half of a ground-truth box). The parameter β is set to 0.25 (arbitrarily) in the original work and we use the same value in our experiments.

This method presents a discontinuity which is common in CP. Either the predictions cover over 75% of ground-truth boxes, or they don't. Thanks to CRC, this discontinuity has been removed and we can directly control the risk itself as the proportion of ground-truth boxes that isn't covered. We also further explore an evolution of this approach that instead considers as risk the average area of ground-truth boxes that isn't covered, which therefore increases the penalty of misdetecting large ground-truth boxes, while reducing it for smaller ones. In order to formulate those losses, let us define some notations.

4.2.1. Losses for Conformal Risk Control

For all calibration samples $(X_i, Y_i)_{i=1,\dots,n_{\text{cal}}}$, let $Y_i \in \{\varnothing, \mathbb{R}^{1\times 4}, \mathbb{R}^{2\times 4}, \dots\}$ be the set of ground-truth bounding boxes included in X_i, which could be empty.

The **box-wise recall** loss is defined as the proportion of boxes that is not entirely covered by prediction boxes. It is formulated as:

$$L_i^{OD-box}(\lambda) = \ell\big(\widehat{\mathcal{C}}_\lambda(X_i), Y_i\big) = \begin{cases} 0 & \text{if } Y_i = \varnothing \\ 1 - \frac{1}{n_i}\sum_k \mathbb{1}_{Y_i^k \subseteq \bigcup_j \widehat{\mathcal{C}}_\lambda(X_i)^j} & \text{otherwise.} \end{cases} \tag{16}$$

This formulation of the loss implies that multiple prediction boxes can be used to cover a single object and be considered correct. There is no discrimination between a single prediction box covering the whole ground-truth box, and hundreds of pixels-sized boxes covering the ground truth too. On the other hand, the pixel-wise recall is designed to tolerate partly covered ground truth, and is smoother than the box-wise recall. It requires further definition as follows.

Let $\mathcal{A}(Y_i^k)$ be the area (in terms of pixels) covered by the box Y_i^k. Moreover, let $Y_i^k \cap \bigcup_j \widehat{\mathcal{C}}_\lambda(X_i)^j$ denote the area of the intersection of the ground-truth box with all predicted boxes. The **pixel-wise recall** loss is then defined as the average proportion of the area of ground-truth boxes that isn't covered by predicted boxes. It is formulated as:

$$L_i^{OD-pixel}(\lambda) = \ell\big(\widehat{\mathcal{C}}_\lambda(X_i), Y_i\big) = \begin{cases} 0 & \text{if } Y_i = \varnothing \\ 1 - \frac{1}{n_i}\sum_k \frac{\mathcal{A}\left(Y_i^k \cap \bigcup^j \widehat{\mathcal{C}}_\lambda(X_i)^j\right)}{\mathcal{A}\left(Y_i^k\right)} & \text{otherwise.} \end{cases} \tag{17}$$

This loss, as the previous ones, tolerates multiple boxes used to cover a single ground truth, even partly in this case. Moreover, this loss is expected to be impacted more by larger ground-truth boxes, as models tend to very rarely predict boxes that are too small for small ground truths. It also can be seen as a further relaxation (smoothing) of the previous loss.

4.2.2. Computation of Prediction Sets

With these elements, we can apply Conformalized Risk Control to a pre-trained OD predictor $\widehat{f}$, provided that we have access to some calibration data. The details are in Algorithm 2.

Algorithm 2: (image-wise) conformally risk-controlled OD: *conformalization*

```
1. Split (disjointly) training data: $D_{\text{train}} = D_{\text{fit}} \uplus D_{\text{cal}}$
2. Fit (or fine-tune) the predictor $\widehat{f}$ on $D_{\text{fit}}$
3. Compute the losses $L_i^{OD}$ on $D_{\text{cal}}$
4. Estimate $\widehat{\lambda}$ as in Equation 4
```

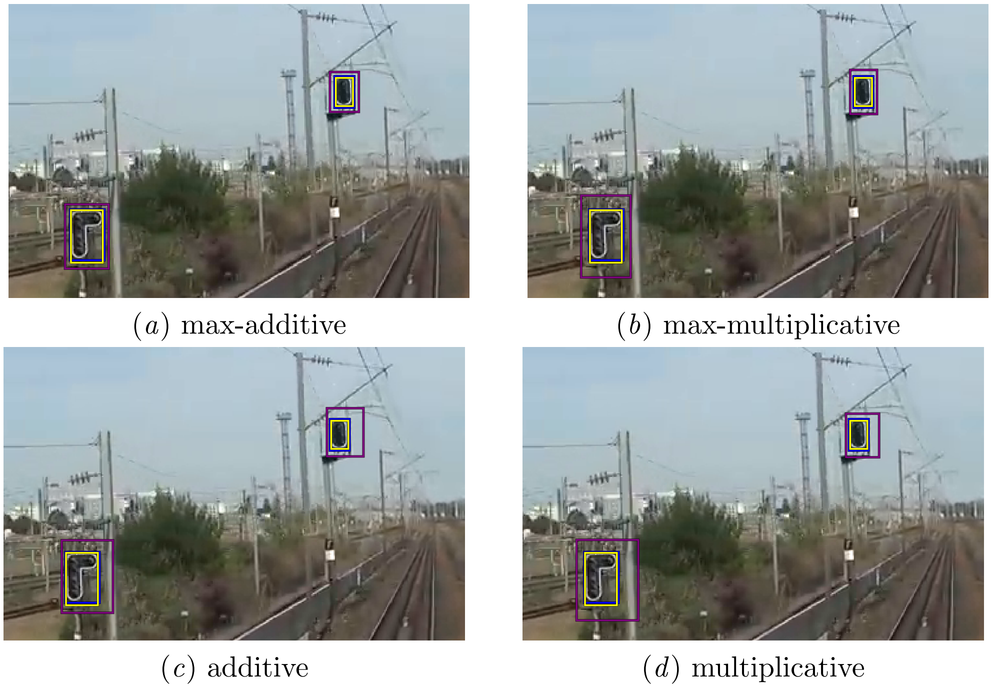
During inference, we build the prediction set identically (although with λ replacing quantiles) as in the box-wise method, for all three conformal methods:

$$\widehat{C}_{\widehat{\lambda}}(X_{n+1}) = \Big\{ \widehat{x}_{\min} - \widehat{\lambda},\ \widehat{y}_{\min} - \widehat{\lambda},\ \widehat{x}_{\max} + \widehat{\lambda},\ \widehat{y}_{\max} + \widehat{\lambda} \Big\}. \quad \text{(CRC additive)} \tag{18}$$

$$\widehat{C}_{\widehat{\lambda}}(X_{n+1}) = \Big\{ \widehat{x}_{\min} - \widehat{w} \cdot \widehat{\lambda},\ \widehat{y}_{\min} - \widehat{h} \cdot \widehat{\lambda},\ \widehat{x}_{\max} + \widehat{w} \cdot \widehat{\lambda},\ \widehat{y}_{\max} + \widehat{h} \cdot \widehat{\lambda} \Big\}. \quad \text{(CRC multiplicative)} \tag{19}$$

5. Experiments

This section elaborates on the experiments performed on our created dataset. Initially, we describe the experimental setup, including the models utilized, fine-tuning process, and conformal parameters. Subsequently, we analyze the experimental outcomes based on two primary metrics: Stretch and Empirical coverage.



(a) max-additive

(b) max-multiplicative

(c) additive

(d) multiplicative

Figure 4: **Box-wise conformalization on an image with traffic signals at different scales.** Bounding boxes as predicted by the DiffusionDet predictor in yellow, conformalized boxes in purple and ground truth in blue. Cropped for readability.

5.1. Setup of experiments

We consider 2 settings for our experiments. One with two data splits for the three selected pretrained OD models, and another with an additional split for fine-tuning these pretrained models. We then apply the conformal procedure on the calibration set, and evaluate results based on metrics. In the literature (Sesia and Candès, 2020) we found that between 10% to 50% of D_{train} are set aside for D_{cal} (no pretrained $\widehat{f}$). For our experiments with pretrained predictors, we split the 3414 data points into 1914 for calibration and 1500 for testing. This resulted in a total of 1974 bounding boxes in D_{test}. For the other set of experiments on fine-tuned predictors, we set aside 1414 points to D_{fit}, 1000 to D_{cal} and 1000 to D_{test}, for a total of 1022 bounding boxes in D_{test}. Throughout the tests, we set $\alpha = 0.1$, the objectness threshold of the predictor to 0.3 and the IoU threshold to 0.3.

5.1.1. Fine-tuning the detectors

As mentioned above, for the second batch of experiments we set aside a partition of data D_{fit} to fine-tune YOLOv5m and DETR-ResNet50. We also attempted fine-tuning of the DiffusionDet model but were eventually unsuccessful. The fine-tuning on the YOLOv5m model was done with its standard learning procedure, on all layers, for 100 epochs with a learning rate of 0.001. For the DETR-ResNet50 model, only 3 epochs of fine-tuning were conducted, optimizing only the 10 top layers of the backbone. The Adam optimizer was used with learning rate 3×10^{-5} and weight decay 10^{-6}. A small amount of data augmentation was added, with brightness, contrast and saturation jittering at level 0.1.

5.1.2. Performance of baseline predictors

Table 2: Comparing models via Average Precision for an IoU threshold ≥ 0.3.

	Model	Average Precision
Pretrained	YOLOv5m	0.23
	DETR-ResNet50	0.29
	DiffusionDet	**0.45**
Finetuned	YOLOv5m	0.36
	DETR-ResNet50	**0.42**

We report in Table 2 the performance in terms of average precision of the multiple models that we chose to use in our experiments, both in their pretrained and finetuned version, as a reference for the evaluation of the different models in terms of conformal prediction performance. We recall that the average precision metric is computed as the area under of recall-precision curve, for recall and precision values computed at different objectness thresholds, i.e. minimum confidence of the model in that its own predicted boxes contain indeed an object. We notice that the pretrained DiffusionDet predictor outperforms all others, including fine-tuned ones, and as the conformal procedure is post-hoc and therefore based on the model's outputs, the obtained quantiles or margins should vary, potentially significantly between models.

5.2. Evaluation metrics

To assess and contrast the various conformalization techniques, we employ two types of metrics in our experiments. The first metric we introduce is called 'stretch', which computes the average ratio of the areas of the conformalized boxes to the area of their corresponding raw prediction boxes. It can be expressed more formally as:

$$\text{Stretch} = \sum_{i=1}^{n} \sum_{j=1}^{n_i} \sqrt{\frac{\mathcal{A}\big(C(X_i)^j\big)}{\mathcal{A}\big(\widehat{f}(X_i)^j\big)}}. \tag{20}$$

This metric is reported respectively for box-wise and image-wise conformalization in Table 3 and Table 5. Moreover, this metric is expected to be biased towards the multiplicative method, as it measure itself a multiplicative coefficient of growth rather than an additive one. The second metric we use is the empirical coverage, or empirical risk. We compute the empirical coverage for CP methods and empirical risk for CRC. These metrics are only useful to measure how close the test performance of the conformalized sets is to the desired one, and higher(or lower) does not imply better. In fact, a coverage significantly higher than the desired level implies conformalized boxes larger than what would have been needed for that application. The coverage is defined as :

$$\sum_i \mathbb{1}_{Y_i \in \mathcal{C}_{\hat{\lambda}}(X_i)}, \tag{21}$$

while for any loss defined in section 4.2, the risk is characterized by:

$$\sum_i \ell\big(\mathcal{C}_{\hat{\lambda}}(X_i), Y_i\big). \tag{22}$$

5.3. Results

5.3.1. Box-wise results

We report in Table 3 and 4 respectively the stretch values and coverage of the multiple models and nonconformity scores we have experimented with.

In Table 3, it appears that, model-wise, the YOLOv5m outperforms other methods (both pretrained and finetuned) as it leads to the smallest stretch in average. It is crucial to bear in mind that the box-wise conformalization method is exclusively applied to the true positives, which are the groud truth boxes that match with a prediction – based on the IoU metric. Therefore, a model that generates fewer prediction may achieve a lower stretch value, provided that the predictions it generates are precise. It is in fact the case as the YOLOv5m model, especially in its finetuned version outputs very few bounding boxes (of the right class) as compared to the DiffusionDet one.

Table 3: **Box-wise**. Average stretch for multiple models and conformalization approaches.

	Model	max-additive	max-multiplicative	additive	multiplicative
Pretrained	YOLOv5m	1.35	1.45	1.45	1.56
	DETR-ResNet50	1.81	1.68	1.96	1.74
	DiffusionDet	1.53	1.53	1.88	1.69
Finetuned	YOLOv5m	1.33	1.35	1.34	1.36
	DETR-ResNet50	1.61	1.70	1.69	1.70

In Table 4 model-wise we observe that the YOLOv5m has *overall* the most calibrated coverage (close to the desired level $\alpha = 0.1$). However, we note that generally the additive and multiplicative margins computed coordinate-wise seem to largely overcover, which is most likely due to the inner conservativeness of the Bonferroni correction. It appears that the DiffusionDet model with the max-additive or max-multiplicative nonconformity scores leads to the most calibrated prediction sets, close to the finetuned YOLOv5m.

In Figure 4 we can appreciate row-wise the effect of the distance (and therefore the size) of detection on the conformalized boxes under the multiple nonconformity scores. The closest sign (a),(c) is affected negatively, while the more distant one (b),(d) is affected

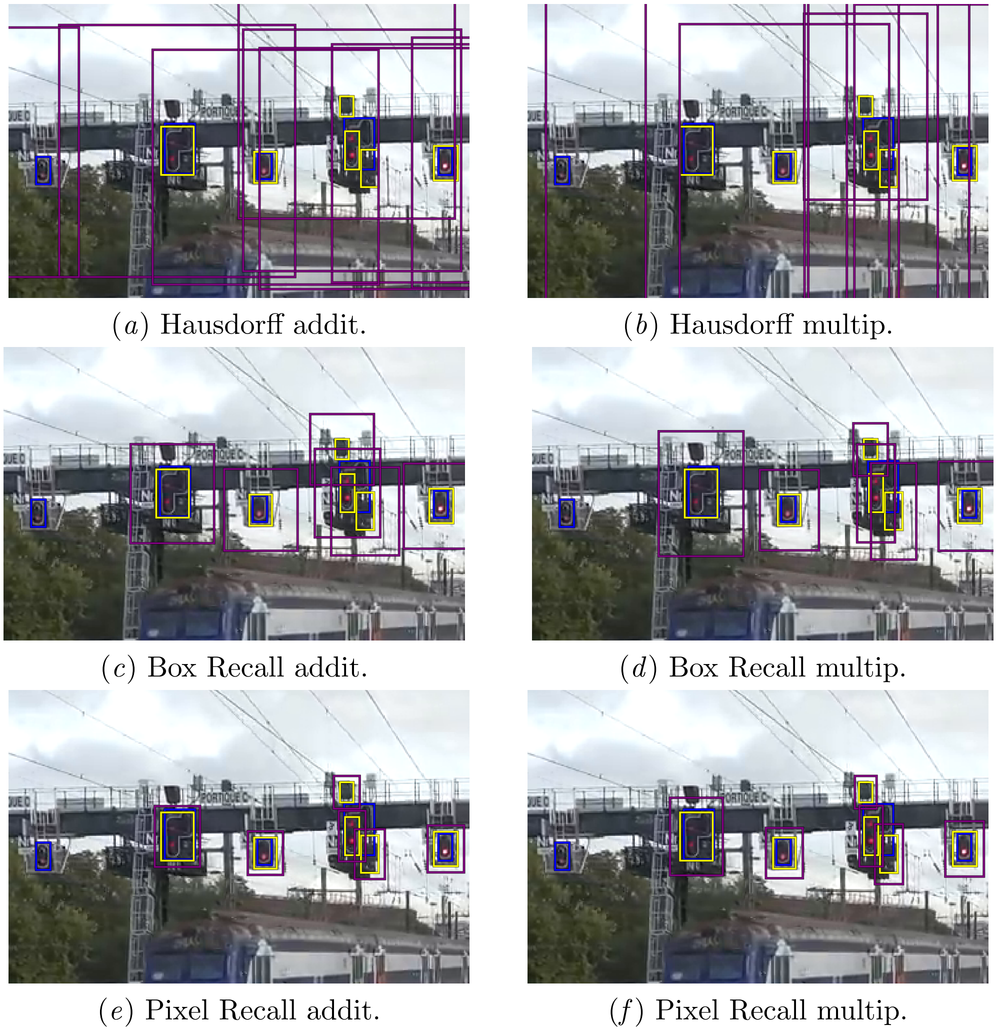


Figure 5: **Image-wise conformalization on an image with several traffic signals (including true and false positives, and false negatives).** Bounding boxes as predicted by the DiffusionDet predictor in yellow, conformalized boxes in purple and ground truth in blue. Cropped for readability.

positively (note that for (b), it is affected positively in terms of width but slightly negatively in terms of height, due to the height being larger than the width and being a factor in the scaling). Column-wise it appears clearly that the max approach has smaller margins than the Bonferroni approach, although it is noticeable that a bias from the model seems to appear, as there is a need for a larger correction in the top and right directions.

5.3.2. Image-wise results

We report in Table 5 and 6 respectively the stretch values, and the coverage/risk values for the different models, correction types (additive or multiplicative) and guarantee type

Table 4: **Box-wise**. Average coverage for multiple models and conformalization approaches.

	Model	max-additive	max-multiplicative	additive	multiplicative
Pretrained	YOLOv5m	0.93	0.94	0.95	0.96
	DETR-ResNet50	0.99	0.98	1.00	0.99
	DiffusionDet	0.92	0.92	0.96	0.96
Finetuned	YOLOv5m	0.94	0.92	0.95	0.93
	DETR-ResNet50	0.95	0.94	0.96	0.95

(conformal, pixel recall or box recall). In Table 6 it is important to note that the coverage is reported for the Hausdorff CP method, while the risk is reported for the CRC methods. As the CP is a generalization of CP, coverage can be considered a (non-smooth) loss, and in that case the risk would be 1 - coverage, as denoted in parenthesis in the table. We recall that better values are the ones close to the desired coverage or risk (resp. 0.9 and 0.1), and significantly better values than desired imply margins larger than necessary. Compared to the box-wise approach, the methods presented here produce well-calibrated prediction sets.

In both tables, the two first lines are missing. This is due to the fact that there is no such value of λ that solves eq. 4. In practice, this is due to a lack of predicted boxes (at the predefined objectness/confidence threshold). Even by increasing the size of predicted boxes infinitely, if there is no predictions for traffic lights by our model on an image, then for all λ, the loss associated with this image will be 1. Therefore for the first two methods, it is not possible to build a prediction set such that the guarantee at level $\alpha = 0.1$ holds.

Table 5: **Image-wise**. Average stretch for multiple models and conformalization approaches. Missing values mean an unattainable risk level, for our predictor and calibration data.

		Hausdorff		Box Recall		Px Recall	
	Model	addit.	multip.	addit.	multip.	addit.	multip.
Pretrained	YOLOv5m	—	—	—	—	—	—
	DETR-ResNet50	—	—	—	—	—	—
	DiffusionDet	9.83	9.69	3.22	2.49	1.56	1.57
Finetuned	YOLOv5m	25.96	22.75	13.06	13.92	12.25	13.22
	DETR-ResNet50	7.74	10.00	3.36	2.75	2.16	2.05

In Table 5, we notice that the YOLOv5m that was best-performing in the box-wise experiments, is the worst performing here. This result is more aligned with expectations according to object detection performances presented in Table 2, and confirms that the performance of YOLOv5m in the previous task was due to the small number and accuracy of its predictions. Moreover, we observe that as the risk, or desired guarantee, is relaxed, the margins on the boxes decrease significantly. Therefore, while the obtained guarantees on the predicted boxes are slightly less strong with the box recall risk than the Hausdorff CP, the conformalized boxes are significantly smaller, and even more so with the pixel recall risk. On Figure 5, appears more clearly than on the table the stretch of the different conformalization approaches. The Hausdorff approach leads to unusable bounding boxes in practice, while the others, in particular (e) lead to very reasonably sized bounding boxes, while holding a guarantee valid on images on average.

Table 6: **Image-wise**. Average coverage and risk for multiple models and conformalization approaches. CRC 0.09: proportion of pixels missed by CRC-conformalized boxes.

		Hausdorff		Box Recall		Px Recall	
	Model	addit.	multip.	addit.	multip.	addit.	multip.
Pretrained	YOLOv5m	—	—	—	—	—	—
	DETR-ResNet50	—	—	—	—	—	—
	DiffusionDet	(1-) 0.90	(1-) 0.91	0.10	0.09	0.09	0.09
Finetuned	YOLOv5m	(1-) 0.90	(1-) 0.90	0.10	0.10	0.10	0.10
	DETR-ResNet50	(1-) 0.87	(1-) 0.87	0.12	0.12	0.12	0.12

5.4. Analysis

A conformalized predictor can only reflect the quality (e.g. accuracy) of its underlying base predictor $\widehat{f}$. If the latter misses many ground-truth boxes, guaranteeing $(1-\alpha)$ 100% correct predictions of a few boxes will still be a small number in the box-wise sense. Furthermore, in the image-wise sense, it will be impossible to reach the desired risk level with an underperforming model (as there may not be any predicted bounding box for a number of images at the predetermined confidence threshold). That is, conformalization is not a substitute for careful training or fine-tuning of a detection architecture, but a complementary tool to increase trustworthiness in the predictive models. For example, we can quantify *how wrong* our predictions are, on average, based on the size of the conformal quantile $q_{1-\alpha}$: multiple predictors can be compared directly against our operational need (coverage, pixel recall, etc.). The interest of capturing the whole box can be operational: our ML pipeline could rely on a conservative estimation of the ground truth to carry out a control operation (e.g. running a ML subcomponent on the detection area).

Concerning the image-wise approach, we noticed very large variations between the different approaches, especially between the Hausdorff and the two others. This is due to a "threshold"effect: margins may be small at a certain level α, but at a slightly lower level $\alpha - \epsilon$, one more ground-truth box has to be included in order to satisfy the guarantee, and in the case the closest box non-covered box is distant, such as the leftmost on Figure 5, the margin can dramatically explode. This effect is increased on the Hausdorff approach, as 75% of boxes need to be covered on 90% on images, while the box-wise approach requires 90% of boxes to be covered in expectation, and therefore can largely fail on difficult images, and compensate on others.

6. Conclusion

Given the insights from this investigation, we intend to develop an enhanced iteration of the dataset, which will serve as a dedicated and high-quality benchmark for evaluating conformal prediction in the domain of object detection, catering to both the scientific community and the transport industry.

It is noteworthy that conformal prediction operates under the assumption of exchangeable data. However, for the deployment of trustworthy AI components in the long run, the problem setting and underlying assumptions will need to be adapted to account for the dynamics of data streams to ensure reliable uncertainty quantification guarantees. This process will present theoretical and practical challenges concerning the construction and validation of datasets.

Our analysis revealed that the current success criterion for prediction relied on the complete coverage of the ground-truth boxes which may be overly restrictive. In practice, it may be adequate for a system to ensure coverage of a substantial portion of the ground truth. This direction deserves more interest from the industrial community, in order to reach much lower risk levels α for viable real-world applications. Moreover, as Split CP only requires little computation at inference, it is easily embeddable in future autonomous systems. Nevertheless, it is improbable that conformal prediction alone can achieve sufficiently low-risk levels. It will be imperative to study larger datasets employing cutting-edge models in tandem with custom-designed conformal methods, and collaborate with domain experts to chart a more definitive course towards certified object detection. Lastly, it is worth noting that several concentration-based approaches have been developed and should be compared in depth to conformal ones.

Acknowledgments

This work has benefited from the AI Interdisciplinary Institute ANITI, which is funded by the French "Investing for the Future – PIA3" program under the Grant agreement ANR-19-P3IA-0004. The authors gratefully acknowledge the support of IRT Saint Exupéry and and the DEEL project (https://www.deel.ai).

References

Lucian Alecu, Hugues Bonnin, et al. Can we reconcile safety objectives with machine learning performances? In *ERTS*, June 2022.

Léo Andéol, Thomas Fel, Florence De Grancey, and Luca Mossina. Conformal prediction for trustworthy detection of railway signals. *arXiv preprint arXiv:2301.11136*, 2023.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends® in Machine Learning*, 16(4):494–591, 2023.

Anastasios N Angelopoulos, Stephen Bates, Emmanuel J Candès, Michael I Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *arXiv preprint arXiv:2110.01052*, 2021.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. *arXiv preprint arXiv:2208.02814*, 2022.

Rina Foygel Barber, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. Predictive inference with the jackknife+. *The Annals of Statistics*, 49(1):486 – 507, 2021.

Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael Jordan. Distribution-free, risk-controlling prediction sets. *J. ACM*, 68(6), 2021.

J. Martin Bland and Douglas G. Altman. Multiple significance tests: the bonferroni method. *BMJ*, 310(6973):170, 1995.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. *arXiv preprint arXiv:2211.09788*, 2022.

Florence de Grancey, Jean-Luc Adam, Lucian Alecu, Sébastien Gerchinovitz, Franck Mamalet, and David Vigouroux. Object detection with probabilistic guarantees: A conformal prediction approach. In *SAFECOMP 2022 Workshops*. Springer, 2022.

Kumari Deepshikha, Sai Harsha Yelleni, PK Srijith, and C Krishna Mohan. Monte carlo dropblock for modelling uncertainty in object detection. *arXiv:2108.03614*, 2021.

Di Feng, Ali Harakeh, Steven L Waslander, and Klaus Dietmayer. A review and comparative study on probabilistic object detection in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2021.

Riccardo Gasparini, Stefano Pini, Guido Borghi, Giuseppe Scaglione, Simone Calderara, Eugenio Fedeli, and Rita Cucchiara. Anomaly detection for vision-based railway inspection. In *EDCC 2020 Workshops*. Springer, 2020.

David Hall, Feras Dayoub, John Skinner, Haoyang Zhang, Dimity Miller, Peter Corke, Gustavo Carneiro, Anelia Angelova, and Niko Sünderhauf. Probabilistic object detection: Definition and evaluation. In *Proceedings of WACV*, pages 1031–1040, 2020.

Ali Harakeh, Michael Smart, and Steven L Waslander. Bayesod: A bayesian approach for uncertainty estimation in deep object detectors. In *Proceedings of ICRA*, 2020.

Jeanine Harb, Nicolas Rébéna, Raphaël Chosidow, Grégoire Roblin, Roman Potarusov, and Hatem Hajri. Frsign: A large-scale traffic light dataset for autonomous trains. *arXiv preprint arXiv:2002.05665*, 2020.

Glenn Jocher. YOLOv5 by Ultralytics, 5 2020. URL github.com/ultralytics/yolov5.

Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *J. of the Am. Stat. Assoc.*, 2018.

Shuo Li, Sangdon Park, Xiayan Ji, Insup Lee, and Osbert Bastani. Towards pac multi-object detection and tracking. *arXiv preprint arXiv:2204.07482*, 2022.

A. Mauri, R. Khemmar, B. Decoux, M. Haddad, and R. Boutteau. Lightweight convolutional neural network for real-time 3d object detection in road and railway environments. *Journal of Real-Time Image Processing*, 19(3):499–516, 2022.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *Proceedings of ECML*. Springer, 2002.

Sangdon Park, Osbert Bastani, Nikolai Matni, and Insup Lee. Pac confidence sets for deep neural networks via calibrated prediction. In *Proceedings of ICLR*, 2020.

Aleksandar Dragan Petrović, Milan Banić, Miloš Simonović, et al. Integration of computer vision and convolutional neural networks in the system for detection of rail track and signals on the railway. *Applied Sciences*, 12(12), 2022.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of CVPR*, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE TPAMI*, 39(6):1137–1149, 2017.

Yaniv Romano, Evan Patterson, and Emmanuel Candes. Conformalized quantile regression. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Matteo Sesia and Emmanuel J Candès. A comparison of some conformal quantile regression methods. *Stat*, 9(1):e261, 2020.

Prashant Singh, Maxim A. Dulebenets, Junayed Pasha, Ernesto D. R. Santibanez Gonzalez, Yui-Yip Lau, and Raphael Kampmann. Deployment of autonomous trains in rail transportation: Current trends and existing challenges. *IEEE Access*, 2021.

Vladimir Vovk. Cross-conformal predictors. *Annals of Mathematics and Artificial Intelligence*, 74(1-2):9–28, July 2013.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, 2nd edition, 2022.

Tao Ye, Zhihao Zhang, Xi Zhang, and Fuqiang Zhou. Autonomous railway traffic object detection using feature-enhanced single-shot detector. *IEEE Access*, 8, 2020.

Oliver Zendel, Markus Murschitz, Marcel Zeilinger, Daniel Steininger, et al. Railsem19: A dataset for semantic rail scene understanding. In *CVPR Workshops*, 2019.

Zhong-Qiu Zhao, Peng Zheng, Shou-Tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE Trans. on Neural Networks and Learning Sys.*, 30(11), 2019.

Arij Zouaoui, Ankur Mahtani, Mohamed Amine Hadded, et al. Railset: A unique dataset for railway anomaly detection. In *Proceedings of IEEE IPAS*, 2022.