Supplementary Material for "On Calibration of Object Detectors: Pitfalls, Evaluation and Baselines"

Selim Kuzucu*, Kemal Oksuz*, Jonathan Sadeghi, and Puneet K. Dokania

Five AI Ltd., United Kingdom
{selim.kuzucu2, kemal.oksuz, jonathan.sadeghi, puneet.dokania}@five.ai

A Further Details on Related Work

Calibration Error in [26] Popordanoska et al. [26] recently proposed Calibration Error that generalise both Detection Expected Calibration Error (D-ECE) and Localisation-aware Expected Calibration Error (LaECE) through a link function $\mathrm{L}(\hat{b}_i, b_{\psi(i)})$ that can be considered as a generic accuracy term. In particular, $\mathrm{L}(\hat{b}_i, b_{\psi(i)})$ measures the similarity between a detection box $\hat{b}_i$ and its corresponding ground truth bounding box $b_{\psi(i)}$. Then, following the conventional calibration literature, the predicted confidence is aimed to be aligned with this link function as the accuracy. Owing to the generic definition of accuracy, it is thus possible recover different calibration error measures with this notion. Specifically, if the link function $\mathrm{L}(\hat{b}_i, b_{\psi(i)})$ is an indicator function that returns true if the detection is a true-positive (TP) and has an Intersection-over-Unions (IoUs) of at least τ, then D-ECE can be recovered. A similar recovery holds for LaECE, when the link function is taken as the IoU, that is $\mathrm{L}(\hat{b}_i, b_{\psi(i)}) = \mathrm{IoU}(\hat{b}_i, b_{\psi(i)})$. Instead of the classical binning approach used in D-ECE and LaECE, the authors define calibration error by utilising a kernel $\mathrm{k}(p_i, p_j)$ to approximate the bins as follows:

$$\hat{CE} = \frac{1}{K}\sum_{c=1}^{K}\sum_{i\in\hat{\mathcal{D}}^c}\frac{1}{|\hat{\mathcal{D}}^c|}\left|\hat{p}_i - \frac{\sum_{j\in\hat{\mathcal{D}},i\neq j}\mathrm{k}(\hat{p}_i,\hat{p}_j)\mathrm{L}(\hat{b}_i, b_{\psi(i)})}{\sum_{j\in\hat{\mathcal{D}},i\neq j}\mathrm{k}(\hat{p}_i,\hat{p}_j)}\right|, \tag{A.1}$$

where the class-wise errors are averaged over to obtain the final calibration performance.

The Components of Localisation-Recall-Precision Error (LRP) Error In Eq. 1 of Sec. 2, we defined LRP Error. While that definition is intuitive as it combines all three types of errors, i.e., precision, recall and localisation errors, into a single measure, these types of errors are not quantified precisely. To address this and provide insight on the detector, Oksuz et al. [23] showed that Eq. 1 can be rewritten alternatively in the following form:

$$\mathrm{LRP} = \frac{1}{\mathrm{N_{FP}} + \mathrm{N_{FN}} + \mathrm{N_{TP}}}\left(\mathrm{w_{Loc}}\mathrm{LRP_{Loc}} + \mathrm{w_{FP}}\mathrm{LRP_{FP}} + \mathrm{w_{FN}}\mathrm{LRP_{FN}}\right), \tag{A.2}$$

*Equal contributions. SK contributed during his internship at Five AI Oxford team.

with the weights $w_{Loc} = \mathrm{N_{TP}}$, $w_{FP} = |\mathcal{D}|$, and $w_{FN} = |\mathcal{G}|$ controlling the contributions of each error type*. Then, denoting the set of all detections $\hat{\mathcal{D}}$, $\mathrm{LRP_{Loc}}$ measures the average localisation error of the TPs detections ($\psi(i) > 0$),

$$\mathrm{LRP_{Loc}} = \frac{1}{\mathrm{N_{TP}}} \sum_{i \in \hat{\mathcal{D}}, \psi(i) > 0} \mathcal{E}_{loc}(i). \tag{A.3}$$

$\mathrm{LRP_{FP}}$ and $\mathrm{LRP_{FN}}$ correspond to the false-positive (FP) and false-negative (FN) rates as the precision and recall errors respectively:

$$\mathrm{LRP_{FP}} = 1 - \frac{\mathrm{N_{TP}}}{|\hat{\mathcal{D}}|} = \frac{\mathrm{N_{FP}}}{|\hat{\mathcal{D}}|} \text{ and } \mathrm{LRP_{FN}} = 1 - \frac{\mathrm{N_{TP}}}{M} = \frac{\mathrm{N_{FP}}}{M}, \tag{A.4}$$

where M is the number of total objects.

B Analyses of Existing Calibration Measures and Further Details

In this section, we first provide further details on D-ECE-style evaluation, which are not included in the main paper due to the space limitation. Then, we provide our analyses on LaECE-style evaluation and Calibration Error (CE)-style evaluation.

B.1 Further Details on D-ECE-style Evaluation

Derivation of Eq. 4 First and foremost, Eq. 2 defines D-ECE as:

$$\text{D-ECE} = \sum_{j=1}^{J} \frac{|\hat{\mathcal{D}}_j|}{|\hat{\mathcal{D}}|} \left| \bar{p}_j - \text{precision}(j) \right|, \tag{A.5}$$

where $\hat{\mathcal{D}}$ and $\hat{\mathcal{D}}_j$ are the set of all detections and the detections in the j-th bin, and $\bar{p}_j$ and precision(j) are the average confidence and the precision of the detections in the j-th bin. With that, precision can be defined as follows:

$$\text{precision}(j) = \frac{\sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i) > 0} 1}{|\hat{\mathcal{D}}_j|}. \tag{A.6}$$

Therefore, Eq. (A.5) can be expressed as:

$$\text{D-ECE} = \sum_{j=1}^{J} \frac{|\hat{\mathcal{D}}_j|}{|\hat{\mathcal{D}}|} \left| \frac{\sum_{\hat{b}_i \in \hat{\mathcal{D}}_j} \hat{p}_i}{|\hat{\mathcal{D}}_j|} - \frac{\sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i) > 0} 1}{|\hat{\mathcal{D}}_j|} \right| \tag{A.7}$$

$$= \frac{1}{|\hat{\mathcal{D}}|} \sum_{j=1}^{J} \left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j} \hat{p}_i - \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i) > 0} 1 \right| \tag{A.8}$$

*Following our design choice in our evaluation framework, here we use $\tau = 0$. Hence, $\mathcal{E}_{loc}(i) = \frac{1 - \mathrm{IoU}(\hat{b}_i, b_{\psi(i)})}{1 - \tau}$ in Eq. 1 reduces to $\mathcal{E}_{loc}(i) = 1 - \mathrm{IoU}(\hat{b}_i, b_{\psi(i)})$ and $w_{Loc} = \mathrm{N_{TP}}$.

Decoupling the errors of TPs ($\psi(i) > 0$) and FPs ($\psi(i) = -1$), and rearranging the terms in the summation, we have

$$\text{D-ECE} = \frac{1}{|\hat{\mathcal{D}}|} \sum_{j=1}^{J} \left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i)>0} \big(\hat{p}_i - 1\big) + \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i)=-1} \hat{p}_i \right|. \tag{A.9}$$

As a result, D-ECE corresponds to the sum of normalized errors in each bin where the normalization constant is the number of detections ($|\hat{\mathcal{D}}|$), and hence minimising

$$\left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i)>0} \big(\hat{p}_i - 1\big) + \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i)=-1} \hat{p}_i \right|, \tag{A.10}$$

minimises D-ECE for the j-th bin, concluding the derivation.

Details of Fig. 3 The calibration errors require to match accuracy of a population (a set of detections) with the average confidence score of the same population where the population is commonly represented as the detections in a bin. That is why, computing a calibration error of the i-th detection, as we did in Fig. 3(d-f), requires some assumptions on other detections except the i-th one. In particular, when we plot a calibration error for a single detection, we assume that the confidence scores of all other detections are constant at the point where the calibration error is minimised for. To illustrate this minimisation criterion, D-ECE is minimized when all TPs have a confidence of 1 and FPs have a confidence of 0 as we showed in Eq. 4. More specifically, we assume that the confidence scores of other detections are equal to their target confidences used to obtain the post-hoc calibrators. To make it more clear, we provide an example below.

As an example, when we plot D-ECE for the TP detection belonging to the car on the left in Fig. 3(b), we assume that the confidence of the FP detection belonging to the car on the right is 0.00 based on Eq. 4. Therefore, the only positive contribution to the D-ECE originates from the detection belonging to the car on the right as the one that we are interested in. Now using the alternative (and equivalent) definition of D-ECE we obtained in Eq. A.9 as

$$\text{D-ECE} = \frac{1}{|\hat{\mathcal{D}}|} \sum_{j=1}^{J} \left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i)>0} \big(\hat{p}_i - 1\big) + \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j, \psi(i)=-1} \hat{p}_i \right|, \tag{A.11}$$

we only focus on the error originating from a single detection, which is the TP car on the right. Specifically, (i) ignoring the normalisation constant $|\hat{\mathcal{D}}|$ and (ii) considering that all other detections contribute to D-ECE with an error of 0 as their confidence matches the target confidence, Eq. (A.11) reduces to

$$|\hat{p}_i - 1| = 1 - \hat{p}_i, \tag{A.12}$$

which is the function we plot in Fig. 3(d) for D-ECE.

Similarly, for D-ECE, the calibration error function of a FP can be derived as $\hat{p}_i$, which is the function we plot for both (e) and (f). We can also obtain the LaECE of a detection for different measures by following the same methodology, that is $|\hat{p}_i - \text{IoU}(\hat{b}_i, b_{\psi(i)})|$ for a TP (as in (d)) and $\hat{p}_i$ for a FP (as in (e) and (f)). For our measures, as $\tau = 0$, we observe that (e) also follows $|\hat{p}_i - \text{IoU}(\hat{b}_i, b_{\psi(i)})|$. As for the discussion why COCO-style D-ECE remains constant in (e), please refer to App. B.3.

Training Detailes of Cityscapes Here we present the implementation details of Cityscapes-style training we used to obtain the results in Tab. 3. As we mentioned, compared to COCO-style training, we make two different modifications:

- Considering the original resolution of the images in Cityscapes dataset [4], which is 2048×1024, we replace the standard augmentation of D-DETR designed for COCO by multi-scale training by resizing the shorter side of the image randomly between $[800, 1024]$ while limiting the longer side with 2048 and keeping the aspect ratio. At inference time, we use the original image resolution, that is 2048×1024.
- Instead of limiting the training of D-DETR by 50 epochs in COCO-style training, we train them all for 200 epochs to bring the number of iterations closer between the training regimes for COCO and Cityscapes. While doing that, we keep the learning rate at $2e-4$ with a learning rate drop to $2e-5$ after 160-th epoch.

B.2 Analyses of LaECE-style Evaluation

LaECE-style evaluation [24] relies on LRP Error (Eq. 1) and LaECE (Eq. 3) for accuracy and calibration respectively. Though we are inspried by [24] for this way of coupling calibration and accuracy, LaECE-style evaluation suffers from critical drawbacks on the informativeness of the confidence scores and the dataset design as we discuss below.

1. Model-dependent threshold selection. As LRP Error is preferred for accuracy and the thresholds are required to be obtained on the val. set, this way of evaluation satisfies this principle.

2. Unambiguous & fine-grained confidence scores. Similar to D-ECE, LaECE also requires FPs to have a confidence of 0.00 regardless of their localisation quality, which might be useful for the subsequent systems. That is why, as illustrated by the right car in Fig. 3(b), a critical information has been missed once $\tau = 0.50$ as its target confidence is set to 0.00 as suggested in [24].

3. Properly-designed datasets. Another critical drawback of this approach is that the in-distribution (ID) test set is obtained from a different distribution. Specifically, the proposed Self-aware Object Detection (SAOD) task in [24] includes two different settings, that are common objects and autonomous vehicle. In both of the cases, the models are evaluated on a test set collected from a different dataset. As an example, Obj45K, as a subset of Objects365 [29] dataset is used to evaluate models trained with COCO. However, as a different

dataset introduces domain shift, the settings for SAOD task cannot be employed to evaluate the calibration performance for ID. By including the Obj45K split, we demonstrate the effect of domain shifted test set on calibration performance in Tab. 6. Specifically, one cannot clearly observe the benefit of post-hoc calibration in Tab. 6 once Obj45K split is used, whereas the post-hoc approaches, which are obtained on ID val. set, improve calibration performance of ID test set significantly. This shows that both ID and domain-shifted test sets should be part of the evaluation, while this is not the case for LaECE-style evaluation.

4. Properly-trained detectors & calibrators. Finally, this way of evaluation does not have a major issue in terms of the used detectors and calibrators. Specifically, four different detectors are used and calibrated with Isotonic Regression (IR) and Linear Regression (LR) post-hoc approaches. Among minor issues, one thing to note is that Platt Scaling, as a distribution calibration approach, has not been investigated in [24]. Furthermore, the applicability of the calibration approaches are not considered from a broader perspective in terms of detectors. In this paper, we design Platt Scaling (PS) properly, and show that PS and IR are quite strong baselines in various scenarios including object detection and instance segmentation using very different detectors.

B.3 Analyses of CE-style Evaluation

CE-style evaluation thresholds the detectors from 0.50 to compute (i) Average Precision (AP) for accuracy; and (ii) CE (Eq. (A.1)) along with D-ECE for calibration. Another peculiarity of this approach is to employ COCO-style CE and D-ECE is the main evaluation measures for calibration performance, which we will provide further details below.

1. Model-dependent threshold selection. As, this type of evaluation also uses a fixed threshold, that is 0.50, the threshold selection is model-independent. Therefore, as the calibration evaluation is quite sensitive to the threshold choice as we showed in Sec .3, this evaluation approach can also lead to the ambiguity on the best performing detector in terms of calibration. In addition, different from D-ECE-style evaluation, the authors also threshold the detection set from 0.50 while computing the accuracy of the detector using AP. However, AP is also quite sensitive to thresholding and can easily mislead the evaluation. To see that, as an example, please consider Fig. 2(b) in which we plot AP of five different detectors for different confidence score thresholds. When we use 0.50, RS R-CNN performs the best, even outperforming Deformable DETR, the most recent detector among all five detectors by around 10 AP points. It also outperforms the recent ATSS by ~ 20 AP once thresholded from 0.50. However, these large gaps in their accuracy only result from the fact that RS R-CNN is more overconfident compared to the other two. When we use our evaluation approach in Tab. 9, we can easily see that D-DETR performs 1.4 LRP better than RS R-CNN and RS R-CNN only outperforms ATSS by 0.8 LRP, instead of 20AP. Therefore, using a fixed threshold on AP does not enable the practitioners to compare the accuracy of the detectors.

2. Unambiguous & fine-grained confidence scores. As we briefly discussed in Sec .3, [26] utilizes COCO-style D-ECE, D-ECE$_{\text{C}}$, and COCO-style CE as the calibration error. Specifically, D-ECE$_{\text{C}}$ (and similarly for CE) is the average of 10 $\text{D}-\text{ECE}$ values that are obtained for different IoU thresholds to validate TPs, i.e., from $\tau = 0.50$ to $\tau = 0.95$ with 0.05 increments. However, as we illustrated in the left car of Fig. 3(b), this way of computing $\text{D}-\text{ECE}$ can result in the same error value regardless of the estimated confidence score for a detection. To demonstrate this, we start with a simpler version of D-ECE$_{\text{C}}$ in which we obtain D-ECE from two IoU thresholds as 0.50 and 0.75 (D-ECE$_{50}$ and D-ECE$_{75}$) and then estimate their average, which can be expressed as:

$$\text{D-ECE}_{\text{C}} = \frac{1}{2}(\text{D-ECE}_{50} + \text{D-ECE}_{75}) \tag{A.13}$$

$$= \frac{1}{2}\sum_{j=1}^{J}\frac{|\hat{\mathcal{D}}_j|}{|\hat{\mathcal{D}}|}\,|\bar{p}_j - \text{precision}_{50}(j)| + \frac{1}{2}\sum_{j=1}^{J}\frac{|\hat{\mathcal{D}}_j|}{|\hat{\mathcal{D}}|}\,|\bar{p}_j - \text{precision}_{75}(j)| \tag{A.14}$$

where we followed the notation from Sec. 2, and precision_{50} and precision_{75} refer to the precision obtained for $\tau = 0.50$ and $\tau = 0.75$ respectively. As we derived in Eq. (A.9) of App. B.1, $\text{D}-\text{ECE}$ can be expressed as the normalized sum of bin-wise errors, hence replacing it for each D-ECE with different thresholds:

$$\frac{1}{2|\hat{\mathcal{D}}|}\left(\sum_{j=1}^{J}\left(\left|\sum_{\substack{\hat{b}_i\in\hat{\mathcal{D}}_j\\ \psi_{50}(i)>0}}(\hat{p}_i-1)+\sum_{\substack{\hat{b}_i\in\hat{\mathcal{D}}_j\\ \psi_{50}(i)=-1}}\hat{p}_i\right|+\left|\sum_{\substack{\hat{b}_i\in\hat{\mathcal{D}}_j\\ \psi_{75}(i)>0}}(\hat{p}_i-1)+\sum_{\substack{\hat{b}_i\in\hat{\mathcal{D}}_j\\ \psi_{75}(i)=-1}}\hat{p}_i\right|\right)\right) \tag{A.15}$$

where $\psi_{50}(i)$ refers to $\psi(i)$ when $\tau = 0.50$, and similarly for 0.75. That is, $\psi_{50}(i) > 0$ implies that i-th detection is a TP for the IoU threshold of $\tau = 0.50$.

COCO-style D-ECE in Eq. (A.15) can yield ambiguous confidence scores for detections with $\psi_{50}(i) > 0$ but $\psi_{75}(i) = -1$, that is a detection with IoU with the object more than 0.50 but less than 0.75. We now demonstrate this on the example below.

Example. We assume that the detector has a single detection with an IoU of 0.60 and compute the COCO-style D-ECE below by exploiting Eq. (A.15):

$$\frac{1}{2}\left(\sum_{j=1}^{J}\left(\left|\sum_{\substack{\hat{b}_i\in\hat{\mathcal{D}}_j\\ \psi_{50}(i)>0}}(\hat{p}_i-1)\right|+\left|\sum_{\substack{\hat{b}_i\in\hat{\mathcal{D}}_j\\ \psi_{75}(i)=-1}}\hat{p}_i\right|\right)\right) \tag{A.16}$$

$$=\frac{1}{2}\left(\left|\sum_{\substack{\hat{b}_i\in\hat{\mathcal{D}}_j\\ \psi_{50}(i)>0}}(\hat{p}_i-1)\right|+\left|\sum_{\substack{\hat{b}_i\in\hat{\mathcal{D}}_j\\ \psi_{75}(i)=-1}}\hat{p}_i\right|\right) \tag{A.17}$$

$$=\frac{1}{2}\left(\left|\hat{p}_i-1\right|+\left|\hat{p}_i\right|\right)=\frac{1}{2}\left(1-\hat{p}_i+\hat{p}_i\right)=0.50 \tag{A.18}$$

Please note that, Eq. (A.16) shows that COCO-style D-ECE results in a constant value that is independent of the predicted confidence score $\hat{p}_i$. This simple example can be easily extended to COCO-style D-ECE with 10 different IoU thresholds for evaluation resulting in the case in the left car in Fig. 3(b). As its IoU with the object is 0.74, it will be considered as a TP by five D-ECE values with $0.50 \leq \tau < 0.75$ and a FP for the other five TP validation thresholds, i.e. $0.75 \leq \tau < 1.00$ in COCO-style D-ECE. Please note that, this also creates ambiguity while we aim to assign targets to the predictions to obtain post-hoc calibrators as the target confidence of such detections is ambiguous unlike the standard D-ECE or LaECE. Considering these drawbacks, we assert that COCO-style computation of calibration errors should be avoided.

3. Properly-designed datasets. As we discussed in Sec. 1, using domain-shifted evaluation sets is crucial for safety-critical applications though they are not used by this way of evaluation.

4. Properly-trained detectors & calibrators. In terms of baseline calibration methods, [26] follows D-ECE-style evaluation and uses Temperature Scaling (TS) as a baseline. As we show in this paper that TS is not the ideal approach for post-hoc calibration of object detectors, [26] also violates this principle.

C Further Details of Our Approach

In this section, we provide further details on our calibration measures, datasets and post-hoc calibrators.

C.1 Further Details on LaECE and Localisation-aware Adaptive Calibration Error (LaACE)

In the following we obtain under which condition LaECE_0 and LaACE_0 are minimized. That is, we show that $\hat{p}_i = \text{IoU}(\hat{b}_i, b_{\psi(i)})$ is a sufficient condition for both measures, while it is also necessary condition for LaACE_0.

The Optimisation Criterion for LaECE_0 For class c, LaECE_0 is defined as:

$$\text{LaECE}_0^c = \sum_{j=1}^{J} \frac{|\hat{\mathcal{D}}_j^c|}{|\hat{\mathcal{D}}^c|} \left| \bar{p}_j^c - \bar{\text{IoU}}^c(j) \right|, \tag{A.19}$$

which can be expressed as,

$$\sum_{j=1}^{J} \frac{|\hat{\mathcal{D}}_j^c|}{|\hat{\mathcal{D}}^c|} \left| \frac{\sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \hat{p}_i}{|\hat{\mathcal{D}}_j^c|} - \frac{\sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \text{IoU}(\hat{b}_i, b_{\psi(i)})}{|\hat{\mathcal{D}}_j^c|} \right|, \tag{A.20}$$

where we replace $\bar{p}_j$, the average of the confidence score in bin j by $\bar{p}_j = \frac{\sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \hat{p}_i}{|\hat{\mathcal{D}}_j^c|}$ and $\bar{\text{IoU}}^c(j)$ by

$$\frac{\sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \text{IoU}(\hat{b}_i, b_{\psi(i)})}{|\hat{\mathcal{D}}_j^c|}. \tag{A.21}$$

Cancelling out $|\hat{\mathcal{D}}_j^c|$ as it is a positive number, we have

$$\text{LaECE}_0^c = \frac{1}{|\hat{\mathcal{D}}^c|} \sum_{j=1}^{J} \left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \hat{p}_i - \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \text{IoU}(\hat{b}_i, b_{\psi(i)}) \right|, \tag{A.22}$$

This implies that the calibration error in bin j is minimized once the following expression is minimized

$$\left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \hat{p}_i - \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \text{IoU}(\hat{b}_i, b_{\psi(i)}) \right| = \left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \hat{p}_i - \text{IoU}(\hat{b}_i, b_{\psi(i)}) \right|, \tag{A.23}$$

implying that LaECE_0 is minimized if $\hat{p}_i = \text{IoU}(\hat{b}_i, b_{\psi(i)})$ for all detections.

The Optimisation Criterion for LaACE_0 For class c, LaACE_0 is defined as:

$$\text{LaACE}_0^c = \sum_{i=1}^{|\hat{\mathcal{D}}^c|} \frac{1}{|\hat{\mathcal{D}}^c|} \left| \hat{p}_i - \text{IoU}(\hat{b}_i, b_{\psi(i)}) \right|, \tag{A.24}$$

As $\frac{1}{|\hat{\mathcal{D}}^c|}$ is a positive constant, LaACE_0^c is simply minimized if and only if $\hat{p}_i = \text{IoU}(\hat{b}_i, b_{\psi(i)})$ for all detections.

$\text{LaACE}_0 \geq \text{LaECE}_0$ **holds.** We now investigate the relationship between LaACE_0 and LaECE_0, and show that LaACE_0 is greater than or equal to LaECE_0, making LaACE_0 a more challenging measure. To show that, we first consider the definition of LaACE_0^c for class c, which is

$$\text{LaACE}_0^c = \sum_{i=1}^{|\hat{\mathcal{D}}^c|} \frac{1}{|\hat{\mathcal{D}}^c|} \left| \hat{p}_i - \text{IoU}(\hat{b}_i, b_{\psi(i)}) \right|, \tag{A.25}$$

which is equal to

$$\text{LaACE}_0^c = \frac{1}{|\hat{\mathcal{D}}^c|} \sum_{j=1}^{J} \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \left| \hat{p}_i - \text{IoU}(\hat{b}_i, b_{\psi(i)}) \right|, \tag{A.26}$$

as we simply the detections into bins but still compute LaACE_0^c by measuring the gap between predicted confidence and IoU for each detection. Now considering

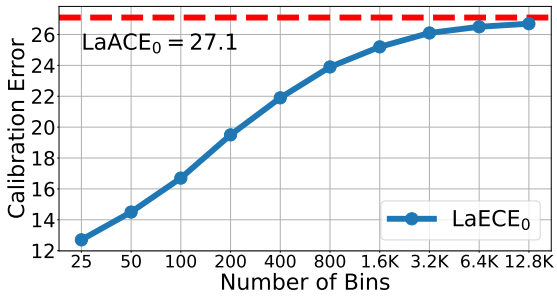


Fig. A.1: LaACE$_0$ (red dashed line at 27.1) and LaECE$_0$ over different number of bins (blue curve) using uncalibrated D-DETR on COCO *minitest*. The number of bins starts from the original 25 bins for LaECE and gets multiplied up by 2 for each step. LaECE$_0$ converges to LaACE$_0$ as the number of bins increases.

the triangle inequality, which is $|x|+|y| \geq |x+y|$, we can take the absolute value out of the inner summation term,

$$\text{LaACE}_0^c \geq \frac{1}{|\hat{\mathcal{D}}^c|} \sum_{j=1}^{J} \left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \hat{p}_i - \text{IoU}(\hat{b}_i, b_{\psi(i)}) \right| \tag{A.27}$$

$$= \frac{1}{|\hat{\mathcal{D}}^c|} \sum_{j=1}^{J} \left| \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \hat{p}_i - \sum_{\hat{b}_i \in \hat{\mathcal{D}}_j^c} \text{IoU}(\hat{b}_i, b_{\psi(i)}) \right| \tag{A.28}$$

$$= \text{LaECE}_0^c \tag{A.29}$$

Please note that we have already derived the last equality in Eq. (A.22), hence enabling us to conclude that LaACE$_0 \geq$ LaECE$_0$ holds. Furthermore, considering the extreme case of this inequality, LaACE$_0$ = LaECE$_0$ holds in the case that the number of bins used to compute LaECE$_0$ is equal to the number of detections. We demonstrate this in Fig. A.1, in which the resulting LaECE$_0$ approximates LaACE$_0$ as the number of bins used to compute LaECE$_0$ increases.

The Cases where There is no Detection From a Class We observed that there can be cases where there is no detection for a class. In such cases, the denominator, $|\hat{\mathcal{D}}^c|$, is 0 in Eq. 6 and Eq. 7, making LaECE$_0$ and LaACE$_0$ undefined. To prevent this, we simply ignore such classes while computing the calibration errors.

C.2 Details of the Datasets Used in our Evaluation Framework

In the following, we provide further details on each of the settings that we used in Tab. 4.

1. Common Objects Setting For this setting, we rely on COCO dataset [16] which is among the most commonly-used object detection benchmarks.

COCO dataset consists of 80 object classes of varying nature. As COCO contains both bounding box and instance mask annotations, we utilise COCO for both of *object detection* and *instance segmentation* in common objects settings.

Training set. We simply use COCO training split with $118K$ images without any modification.

Validation and ID test sets. As the annotations of the COCO test set are not public, we randomly split the validation set of COCO as minival and minitest sets following the literature [24, 25]. Specifically, both of these sets contain $2.5K$ images, contain objects from each classes in COCO dataset and represent similar characteristics. As an example, while COCO minival contains 7.5 object annotations per image, COCO minitest has 7.2 object annotations.

Domain-shifted test sets. With the aim of providing more comprehensive insights regarding the accuracy and the calibration of the detectors, we also evaluate them under certain corruptions. Specifically, following Oksuz et al. [24], we consider 15 benchmark corruptions from the corruptions provided in [11]. These corruptions can further be listed as *gaussian noise, shot noise, impulse noise, speckle noise, defocus blur, motion blur, gaussian blur, snow, frost, fog, brightness, contrast, elastic transform, pixelate* and *jpeg compression*. Furthermore, we only consider each corruption at the severity levels 1, 2 and 3 as it was previously observed that higher severity levels can alter the semantics of the images, especially resulting in some small objects to disappear [24]. As a result, we report the average LRP, LaECE and LaACE values over 45 different corruption settings (15 corruptions under 3 severities.), providing a comprehensive evaluation. Moreover, for more realistic domain shift, we also borrow Obj45K set [24] which has the same label space with COCO. Specifically, Obj45K contains 45K images with 6.0 object annotations per image. Even though the label space is the same for both of the COCO minitest and Obj45K datasets, there is still a shift between these datasets as they are obtained from different datasets. We use Obj45K only for object detection as this dataset does not have mask labels for instance segmentation.

2. Autonomous Driving Setting Cityscapes [4] is a well-known autonomous driving dataset consisting of 8 classes, namely *person*, *rider*, *car*, *truck*, *bus*, *train*, *motorbike* and *bicyle*. Cityscapes is further used as a common benchmark in the contemporary training-time calibration works for object detection [19–21]. To provide richer insights across multiple application domains, we also report results on the Cityscapes dataset for *object detection*.

Training set. We directly use the Cityscapes training split with 2975 images without any modifications.

Validation and ID test sets. As the annotations of the Cityscapes test set are not public either, we randomly split the validation set of Cityscapes as minival and minitest sets following our common objects setting. Specifically, both of these sets contain 250 images, include objects from all of the classes in the

Cityscapes dataset and show similar characteristics. To exemplify, Cityscapes minival contains 20.7 object annotations while Cityscapes minitest contains 21.0 object annotations.

Domain-shifted test sets. We directly follow the methodology described in COCO minitest-C to construct the Cityscapes minitest-C as the corrupted domain shift set for the autonomous driving setting. Moreover, for a more realistic domain shift setting, we also consider Foggy Cityscapes [28] dataset, consisting of the images in the test set but with an additional realistic fog simulation. Specifically, there are three different fog severity levels presented in Sakaridis et al. [28], one for each of 600m, 300m and 150m meteorological optical ranges (visibility). During our experiments with Foggy Cityscapes, we select the subset of images that are also present in the Cityscapes minitest dataset to preserve the consistency, yielding a total of 250 images for each of the three visibility ranges. We then report the performance by averaging the performance measure over all 3 visibility ranges.

3. Long-tailed Objects Setting Large Vocabulary Instance Segmentation (LVIS) [8] dataset contains over 1000 object classes with rich bounding box and instance mask annotations for *object detection* and *instance segmentation* tasks. Owing to its extremely diverse label set, LVIS comes across as a challenging long-tailed dataset with many rare classes. For all of our settings, we utilise LVIS v1.0 which builds up on the images of COCO while introducing rich and much more diversified annotations.

Training set. We directly use the LVIS v1.0 training split with $100K$ samples without any modifications.

Validation and ID test sets. Similarly with the common objects and autonomous driving settings, we split the LVIS v1.0 validation set into minival and minitest sets. Specifically, both of these sets contain approximately 9.8K images and show similar characteristics. To exemplify, LVIS minival set contains 12.6 object annotations per image and LVIS minitest set contains 12.4 object annotations per image. To enable that post-hoc calibrators are trained properly, we ensure that each class in LVIS minitest is also represented in LVIS minival while we split LVIS val. set into two. As some of the classes have very few instances in the val. set (which is only 1 instance for some classes) due to the long-tailed nature of the dataset, this resulted in a case where LVIS minitest set only includes 935 classes and minival set contains 1035 classes. Accordingly, we evaluate the models on those 935 classes for our long-tailed object setting.

Domain-shifted test sets. To obtain LVIS minitest-C, we follow our approach used for constructing COCO minitest-C and evaluating it.

C.3 Details of Post-hoc Calibrators

This section presents the details of post-hoc calibration methods.

Algorithm A.1 Training calibrator on $\mathcal{D}_{\text{Val}}$

1: **procedure** TRAINCALIBRATOR($\mathcal{D}_{\text{Val}}$)
2: Cross-validate calibration thr. $\bar{u}^c$ for each class on $\mathcal{D}_{val}$ using LRP with $\tau = 0$
3: Remove detections with score less than $\bar{u}^c$ in $\mathcal{D}_{val}$ to obtain $\mathcal{D}_{thr}$
4: Train calibrator $\zeta^c(\cdot)$ for each class c on $\mathcal{D}_{thr}$
5: Calibrate the detections in $\mathcal{D}_{val}$ using $\{\zeta^c(\cdot)\}$ to obtain $\mathcal{D}_{cal}$
6: Cross-validate operating thr. $\bar{v}^c$ for each class on $\mathcal{D}_{cal}$ using LRP with τ
7: **return** $\{\bar{u}^c, \bar{v}^c, \zeta^c(\cdot)\}_{c=1}^{K}$
8: **end procedure**

Algorithm A.2 Calibrating detections from an image

1: **procedure** CALIBRATE($\{\hat{c}_i, \hat{b}_i, \hat{p}_i\}^N$, $\{\bar{u}^c, \bar{v}^c, \zeta^c(\cdot)\}_{c=1}^{K}$)
2: Remove detections with score less than $\bar{u}^c$ in $\{\hat{c}_i, \hat{b}_i, \hat{p}_i\}^N$ to obtain $\mathcal{D}_{thr}$
3: Calibrate confidence scores in $\mathcal{D}_{thr}$, i.e., $\hat{p}_i := \zeta^{\hat{c}_i}(\hat{p}_i)$
4: Remove detections with calibrated score less than $\bar{v}^c$ in $\mathcal{D}_{thr}$
5: **return** remaining detections
6: **end procedure**

Calibrator Training and Inference Algorithms The details of training and inference with a calibrator are in Alg. A.1 and A.2 respectively. In both of the algorithms, we follow the notation that we introduced in Sec. 2 and Sec. 4.4. Also as an extreme case in which no detection remains after thresholding for a class to train the calibrator ($|\mathcal{D}_{thr}| = 0$ in Line 4 of Alg. A.1), we simply use identity function as the calibrator.

Negative Log-Likelihood (NLL) Derivation for Platt Scaling We now aim to minimize the NLL of the predicted calibrated confidence scores ($\hat{p}_i^{cal}$) by considering the target Bernoulli distribution, that is $\mathcal{B}(\text{IoU}(\hat{b}_i, b_{\psi(i)}))$. Specifically, using the standard iid assumption, the likelihood of predicted L calibrated probabilities considering the Bernoulli distribution can be expressed as:

$$\prod_{i=1}^{L} \hat{p}_{cal,i}^{\text{IoU}(b_i, b_{\psi(i)})} (1 - \hat{p}_{cal,i})^{1-\text{IoU}(b_i, b_{\psi(i)})} \tag{A.30}$$

where we use $\hat{p}^{cal,i}$ as $\hat{p}_i^{cal}$ for better readability of the notation. Taking the logarithm and multiplying with -1 to make it negative, we have the following expression to minimize:

$$-\sum_{i=1}^{L} \text{IoU}(b_i, b_{\psi(i)}) \log(\hat{p}_{cal,i}) + (1 - \text{IoU}(b_i, b_{\psi(i)}) \log(1 - \hat{p}_{cal,i}). \tag{A.31}$$

Therefore, the NLL of the i-th example is:

$$-(\text{IoU}(\hat{b}_i, b_{\psi(i)}) \log(\hat{p}_{cal,i}) + (1 - \text{IoU}(\hat{b}_i, b_{\psi(i)})) \log(1 - \hat{p}_{cal,i})), \tag{A.32}$$

which is the cross entropy loss function in Eq. 9.

D Further Experiments

This section presents further experiments and analyses that are not included in the paper.

D.1 Implementation Details

Detectors used for Common Objects Setting We do not train any detector for this setting and use existing detectors. Specifically, for the five training-time calibration methods for object detection in Tab. 5 and Tab. 5, we borrow the detectors in the official repositories of Cal-DETR and BPC. Please note that Cal-DETR repository releases all four detectors except BPC. We also note that among these five approaches, MbLS and MDCA are specifically designed for the classification task, hence their extension to detection are not investigated, and they are used as baselines for TCD, BPC and Cal-DETR. In the same tables and Tab. 8, our baseline D-DETR is taken from mmdetection as we rely on this framework which provides the trained models of several different object detectors. As for Tab. 9, we again use mmdetection with the exceptions of: (i) state-of-the-art (SOTA) detectors and UP-DETR, which we use their official repositories and (ii) MoCaE which we implement ourselves while fully adhering to the original settings including all of the hyperparameters described in [25]. Finally, we again obtain the models mmdetection for instance segmentation task on Tab. A.7 and Tab. A.8, in which we use a Resnet-50 [10] backbone for all the detectors.

Detectors used for Autonomous Driving Setting For this setting, we train all detectors in Tab. 7 and Tab. A.1 as the detectors are not publicly released for this setting. While doing that we keep all hyperparameters for each detector as it is but only make two changes that we outlined in App. B in the training pipeline to boost the performance of the models and compare them properly. Specifically, we incorporate this setting into official repositories of Cal-DETR and BPC, and implemented TCD by ourselves as its implementation with D-DETR is not publicly available. Similarly, please note that, for MbLS and MDCA, two baselines borrowed from classification, their implementation is also not publicly available. Also considering that extending these methods for object detection requires a thourough thought process and diligent hyperparameter tuning, we do not use these baseline for our autonomous driving setting. As for D-DETR, we use mmdetection framework.

Detectors used for Long-tailed Objects Setting Similar to the common objects setting, we simply use trained models from mmdetection for this setting. Specifically, for all three models we used in Tab. A.2, Tab. A.3 and Tab. A.4 for long-tailed detection, we utilise the ones with Resnet-101 backbone.

D.2 Comparison with SOTA in terms of Other Evaluation Approaches on Autonomous Driving Setting

In Tab. 5, we presented that our post-hoc calibrators outperform all existing training-time calibration methods significantly in terms of four different evalua-

Table A.1: Comparison with SOTA methods in terms of other evaluation measures on Cityscapes minitest set. LRP is reported on LRP-optimal thresholds obtained on val. set. AP is reported on top-100 detections. τ is taken as 0.50. All measures are lower-better, except AP. **Bold:** the best, underlined: second best. PS: Platt Scaling, IR: Isotonic Regression.

Cal. Type	Method	Calibration (thr. 0.30) D-ECE	Calibration (thr. 0.30) LaECE	Calibration (LRP thr.) D-ECE	Calibration (LRP thr.) LaECE	Accuracy LRP	Accuracy AP ↑
Uncal.	D-DETR [34]	3.2	20.8	3.5	20.0	68.4	37.3
Training Time	TCD [20]	30.9	18.9	31.5	18.3	70.5	34.2
	BPC [19]	8.3	26.8	9.2	24.9	74.7	30.7
	Cal-DETR [21]	3.7	21.4	3.4	19.9	68.1	37.0
Post-hoc (Ours)	PS for D-ECE	2.8(+0.4)	20.2	2.3(+1.2)	19.8	68.4	37.3
	PS for LaECE	14.2	11.3(+7.6)	14.1	**9.4**(+7.9)	68.4	37.3
	IR for D-ECE	**1.5**(+1.7)	19.6	**1.4**(+2.0)	19.4	68.4	36.8
	IR for LaECE	14.2	**10.4**(+8.5)	14.2	**9.4**(+7.9)	68.4	36.6

Table A.2: Calibrating and evaluating instance segmentation methods on Long-tailed Objects setting based on LVIS. Results are reported on LVIS minitest, please refer to App. D for the domain-shifted LVIS minitest-C. **Bold:** the best calibration approach.

	Object Detection							Instance Segmentation						
	Uncalibrated			Isotonic Regression			Box	Uncalibrated			Isotonic Regression			Mask
Detector	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaACE_0$	LRP	AP ↑	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaACE_0$	LRP	AP ↑
Mask R-CNN [9]	25.2	30.4	74.7	**17.1**	**28.0**	74.6	27.1	25.5	30.6	75.3	**17.6**	**28.4**	75.3	25.9
Seesaw Mask R-CNN [30]	25.0	30.2	73.1	**16.8**	**27.8**	73.0	31.8	25.0	30.2	73.7	**16.7**	**27.6**	73.7	31.0
Seesaw Cascade R-CNN [30]	26.4	31.5	70.7	**17.4**	**28.7**	70.8	36.0	25.5	30.6	71.7	**17.2**	**28.1**	71.6	33.1

tion approaches using existing measures. Please refer to Sec. 5 for further details on these evaluation approaches. We now show that our observations also apply to the autonomous driving dataset. Specifically, in Tab. 5, PS and IR outperform all existing training methods as well as improve the calibration performance of baseline D-DETR on the existing evaluation approaches.

D.3 Calibration Under Long-tailed Class Distribution

As common baselines used for LVIS dataset, here we use Mask R-CNN [9] and Cascade Mask R-CNN [1] along with their stronger versions trained with Seesaw Loss [30]. We obtain two calibrators for each class using the held-out LVIS *minival*: (i) for object detection using the IoU; and (ii) for instance segmentation using mask IoU as the calibration target. Tab. A.2 shows that IR improves calibration by up to 9 $LaECE_0$ and 2.8 $LaACE_0$, showing that it is still a strong baseline for this challenging setting with around 1K classes. However, $LaECE_0$ and $LaACE_0$ are greater compared to COCO (Tab. 6) and Cityscapes (Tab. 7) *minitest*, suggesting the need for further research on calibration under long-tailed data.

Tab. A.3 and Tab. A.4 show the calibration results for PS, our other calibrator, demonstrating that PS also improves calibration performance by more than

Table A.3: Calibrating and evaluating different object detectors for *object detection* on the LVIS minitest dataset. **Bold:** the best, underlined: second best among calibration approaches for each task.

Detector	Uncalibrated			Platt Scaling			Isotonic Regression		
	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP
Mask R-CNN [9]	25.2	30.4	74.7	18.2	28.4	74.7	**17.1**	**28.0**	74.6
Seesaw Mask R-CNN [30]	25.0	30.2	73.1	18.4	28.3	73.1	**16.8**	**27.8**	73.0
Seesaw Cascade R-CNN [30]	26.4	31.5	70.7	19.0	28.8	70.7	**17.4**	**28.7**	70.8

Table A.4: Calibrating and evaluating different object detectors for *instance segmentation* on the LVIS minitest dataset. **Bold:** the best, underlined: second best among calibration approaches for each task.

Detector	Uncalibrated			Platt Scaling			Isotonic Regression		
	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP
Mask R-CNN [9]	25.5	30.6	75.3	18.6	28.6	75.3	**17.6**	**28.4**	75.3
Seesaw Mask R-CNN [30]	25.0	30.2	73.7	18.0	27.8	73.7	**16.7**	**27.6**	73.7
Seesaw Cascade R-CNN [30]	25.5	30.6	71.7	18.4	28.2	71.7	**17.2**	**28.1**	71.6

7 $LaECE_0$ and around 2.5 $LaACE_0$. As a result, PS can also be used as a strong baseline on this challenging dataset.

Furthermore, for the sake of completeness, we now evaluate the aforementioned three detectors of the long-tailed setting under domain shift on LVIS minitest-C. Similarly with Tab. 6 and Tab. 7, IR and PS share the top-2 entries on both the object detection (Tab. A.5) and the instance segmentation (Tab. A.6) settings while preserving the accuracy of the models. As an example, IR improves the $LaECE_0$ of the models up to 8.4 in the object detection and up to 7.6 in the instance segmentation on LVIS minitest-C. These results highlight then even under a domain shifted version of a challenging long-tailed dataset, both of IR and PS remain still quite effective in improving the calibration of the model.

D.4 Instance Segmentation on Common Objects Setting

In addition to evaluating the object detectors with common objects, we now evaluate instance segmentation models. To calibrate these models, we use mask IoU as the target for our calibration measures. For our experiments in this setting, we utilise three well-known detectors, namely HTC [2] , Queryinst [7], and Mask2Former [3]. Tab. A.7 presents the results on COCO minitest, where we can observe that our IR improves the $LaECE_0$ of the models significantly by up to 23.8 and $LaACE_0$ by up to 9.9. PS generally perform similar to IR in terms of $LaACE_0$ but slightly worse on $LaECE_0$. Analogously with the ID test set, we observe drastic calibration improvements with our IR for domain-shifted test set in Tab. A.8, in which it improve $LaECE_0$ up to 22.6. These results further validate the effectiveness of our IR and PS on instance segmentation task.

Table A.5: Calibrating and evaluating different object detectors for *object detection* on the LVIS minitest-C domain shift dataset. **Bold:** the best, underlined: second best among calibration approaches for each task.

Detector	Uncalibrated			Platt Scaling			Isotonic Regression		
	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP
Mask R-CNN [9]	25.9	30.7	83.8	19.8	**28.6**	83.8	**18.9**	28.8	83.8
Seesaw Mask R-CNN [30]	26.3	30.7	83.3	20.0	**28.3**	83.3	**19.0**	28.6	83.3
Seesaw Cascade R-CNN [30]	27.9	32.3	82.0	20.5	**28.9**	82.0	**19.5**	29.3	82.0

Table A.6: Calibrating and evaluating different object detectors for *instance segmentation* on the LVIS minitest-C domain shift dataset. **Bold:** the best, underlined: second best among calibration approaches for each task.

Detector	Uncalibrated			Platt Scaling			Isotonic Regression		
	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP
Mask R-CNN [9]	25.6	30.3	84.4	19.4	**28.1**	84.4	**18.7**	28.4	84.4
Seesaw Mask R-CNN [30]	25.7	30.1	83.9	19.7	**27.9**	83.9	**18.8**	28.4	83.9
Seesaw Cascade R-CNN [30]	26.6	30.9	82.8	19.8	**28.1**	82.8	**19.0**	28.5	82.8

D.5 Further Analyses

Further Ablations Similar to Tab. 8, we perform ablations over different design choices for IR in Tab. A.9 using both COCO-minitest and Cityscapes-minitest. As is the case with TS, domain-shifted val. set degrades the accuracy of the detector, in red font, as the operating thresholds obtained on these val. sets do not generalize to the ID test set. Our final design with thresholding and class-wise calibrators reaches the best or second best performance in terms of all calibration measures, validating our design choice on post-hoc calibrators.

Furthermore, we analyse the behavior of our PS and IR under different design choices for D-ECE as a different calibration measure in Tab. A.10. We note that, as a fixed threshold is not a good approach for evaluation, here we compute D-ECE using LRP-optimal thresholding similar to computing our calibration measures. Accordingly, as discussed in Sec. 4.4, we construct target and prediction pairs to train calibrators by considering D-ECE instead of our localisation-based calibration measures. Tab. A.10 shows that our design choices also generalize to D-ECE as using ID val. set, thresholding the detections and using a bias term either perform the best or the second best in terms of D-ECE also by preserving the accuracy of the detector. As an example, bias term significantly helps for calibration for COCO minitest, reaching 2.4 D-ECE decreasing it from 10.0 compared to not using bias term. These results also validates our design choices.

More Reliability Diagrams In this part, we further provide reliability diagrams for three models from Tab. 9, namely: (i) UP-DETR in Fig. A.2; (ii) EVA [6] in Fig. A.3; and (iii) RS R-CNN in Fig. A.4. The improvements provided

Table A.7: Calibrating and evaluating different object detectors for *instance segmentation*. We use our common objects setting and report the results on COCO *minitest*. **Bold:** the best, underlined: second best among calibration approaches for each task.

Detector	Uncalibrated			Platt Scaling			Isotonic Regression		
	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP
HTC [2]	26.2	29.1	60.5	10.2	23.2	60.5	**7.8**	**22.3**	60.5
QueryInst [7]	11.5	23.8	56.4	10.0	22.8	56.4	**8.2**	**21.9**	56.4
Mask2Former [3]	31.3	32.1	54.1	9.6	22.4	54.1	**7.5**	**22.2**	54.2

Table A.8: Calibrating and evaluating different object detectors for *instance segmentation* under domain shift. We use our common objects setting and report the results on COCO minitest-C domain shift dataset. **Bold:** the best, underlined: second best among calibration approaches for each task.

Detector	Uncalibrated			Platt Scaling			Isotonic Regression		
	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP	$LaECE_0$	$LaECE_0$	LRP
HTC [2]	26.8	30.0	73.9	12.6	24.8	73.9	**10.5**	**24.4**	73.9
QueryInst [7]	13.0	25.2	69.9	11.8	24.2	69.9	**9.8**	**23.5**	70.0
Mask2Former [3]	32.8	33.4	67.8	12.3	**24.1**	67.8	**10.2**	24.3	67.8

by our PS and IR are evident in the reliability diagrams as well in line with the results of Tab. 9.

Comparing the Detectors in Fig. 1 We used five uncalibrated detectors (marked with * in Tab. 9) in Fig. 1 to illustrate how challenging evaluating the object detectors are. We now compare these detectors using our evaluation framework. Tab. 9 shows that D-DETR performs the best whereas Faster R-CNN performs the worst in terms of both accuracy (57.3 vs. 60.4 LRP) and calibration (12.7 vs. 27.0 $LaECE_0$). This is an expected result as (i) D-DETR and Faster R-CNN are trained with Focal Loss [15] and Cross-entropy loss, between which Focal Loss provides better calibration [18]; and (ii) D-DETR is more accurate than Faster R-CNN [34].

Comparing different detectors across all of the existing calibration error measures Not only our observations on the calibration improvements brought in by IR and PS hold on LaECE, we also observe the same pattern of improvement in terms of D-ECE across the same wide range of detectors. The results can be observed from Tab. A.11.

References

1. Cai, Z., Vasconcelos, N.: Cascade R-CNN: Delving into high quality object detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
2. Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., Loy, C.C., Lin, D.: Hybrid task cascade for instance segmentation.

Table A.9: Ablation experiments on Isotonic Regression using D-DETR. **Bold:** the best, underlined: second best. ✗ denotes that a domain-shifted val. set is used for obtaining thresholds and calibration, resulting in a big drop in accuracy (red font).

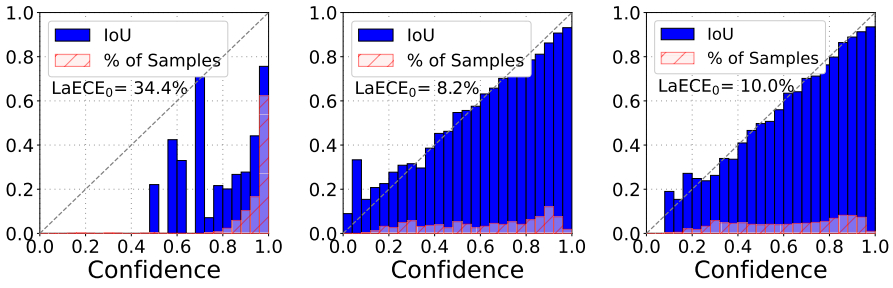
Method	Ablations on Dataset		Ablations on Model	COCO *minitest*			Cityscapes *minitest*		
	ID Val. Set	Threshold	Class-wise	$LaECE_0$	$LaACE_0$	LRP	$LaECE_0$	$LaACE_0$	LRP
Isotonic Regression	✗			14.6	25.1	61.2	10.5	**23.0**	60.2
	✓			10.3	23.6	57.1	12.1	25.8	57.5
	✓	✓		9.8	24.0	57.2	11.3	26.1	57.2
	✓		✓	**7.5**	23.2	58.0	**8.6**	23.8	56.2
	✓	✓	✓	7.7	**23.1**	57.2	9.0	23.7	56.8

Table A.10: Ablation experiments on post-hoc calibrators using D-DETR. **Bold:** the best, underlined: second best. ✗ denotes that a domain-shifted val. set is used for obtaining thresholds and calibration, resulting in a big drop in accuracy (red font). Bias term only exists in the formulation of PS (b in Eq. 8), hence it is N/A for IR.

Method	Ablations on Dataset		Model	COCO *minitest*		Cityscapes *minitest*	
	ID Val. Set	Threshold	Bias Term	D-ECE	LRP	D-ECE	LRP
Platt Scaling	✗			8.0	69.8	**2.7**	71.7
	✓			10.0	66.3	4.3	68.4
	✓	✓		10.0	66.3	3.6	68.4
	✓	✓	✓	**2.4**	66.3	3.2	68.4
Isotonic Regression	✗		N/A	13.2	69.4	6.1	71.9
	✓		N/A	**1.5**	66.0	0.8	68.7
	✓	✓	N/A	2.6	66.2	**0.4**	68.4

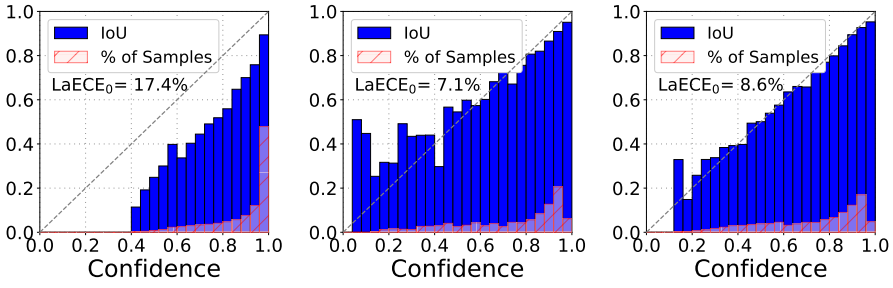
In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

3. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
4. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
5. Dai, Z., Cai, B., Lin, Y., Chen, J.: Unsupervised pre-training for detection transformers. IEEE Transactions on Pattern Analysis and Machine Intelligence p. 1–11 (2022). https://doi.org/10.1109/tpami.2022.3216514, http://dx.doi.org/10.1109/TPAMI.2022.3216514
6. Fang, Y., Wang, W., Xie, B., Sun, Q., Wu, L., Wang, X., Huang, T., Wang, X., Cao, Y.: Eva: Exploring the limits of masked visual representation learning at scale. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
7. Fang, Y., Yang, S., Wang, X., Li, Y., Fang, C., Shan, Y., Feng, B., Liu, W.: Instances as queries. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 6910–6919 (October 2021)



(a) Uncalibrated UP-DETR **(b)** UP-DETR with IR **(c)** UP-DETR with PS

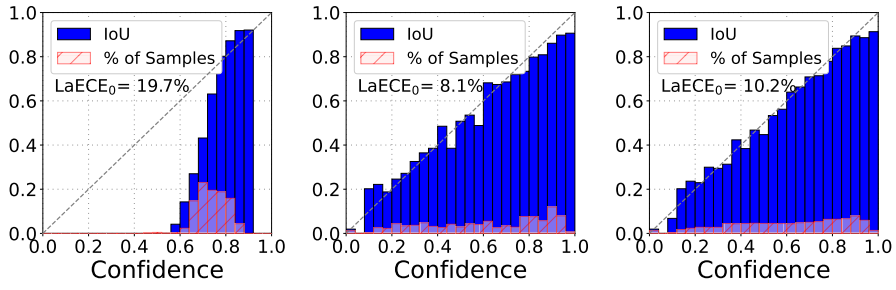
Fig. A.2: Uncalibrated UP-DETR [5] (a), calibrated UP-DETR [5] with isotonic regression (b) and calibrated UP-DETR [5] with platt scaling (c) reliability diagrams on COCO *minitest* [16].



(a) Uncalibrated EVA **(b)** EVA with IR **(c)** EVA with PS

Fig. A.3: Uncalibrated EVA [6] (a), calibrated EVA [6] with isotonic regression (b) and calibrated EVA [6] with platt scaling (c) reliability diagrams on COCO *minitest* [16].

8. Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
9. He, K., Gkioxari, G., Dollar, P., Girshick, R.: Mask R-CNN. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2017)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
11. Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. In: International Conference on Learning Representations (ICLR) (2019)
12. Kim, K., Lee, H.S.: Probabilistic anchor assignment with iou prediction for object detection. In: The European Conference on Computer Vision (ECCV) (2020)
13. Li, L.H., Zhang, P., Zhang*, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., Chang, K.W., Gao, J.: Grounded language-image pre-training. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)



(a) Uncalibrated RS R-CNN **(b)** RS R-CNN with IR **(c)** RS R-CNN with PS

Fig. A.4: Uncalibrated RS R-CNN [22] (a), calibrated RS R-CNN [22] with isotonic regression (b) and calibrated RS R-CNN [22] with platt scaling (c) reliability diagrams on COCO *minitest* [16].

Table A.11: Calibrating and evaluating different object detectors in terms of different calibration errors. We use Common Objects setting and report the results on COCO *minitest*. * denotes the detectors in Fig. 1. Among these detectors, considering the uncalibrated columns, our evaluation ranks the D-DETR as the best. **Bold:** the best, underlined: second best for calibration.

Type	Detector	Backbone	Uncalibrated				Platt Scaling				Isotonic Regression			
			LaECE$_0$	LaACE$_0$	LaECE	D-ECE	LaECE$_0$	LaACE$_0$	LaECE	D-ECE	LaECE$_0$	LaACE$_0$	LaECE	D-ECE
One-Stage	PAA [12]*	R50	15.9	28.1	14.4	27.7	9.7	24.3	9.9	0.9	7.7	23.8	**7.9**	**0.8**
	ATSS [33]*	R50	19.1	34.0	19.8	33.1	10.3	24.7	9.7	**0.3**	8.5	24.1	**8.3**	0.5
	GFL [14]	R50	13.7	28.5	12.9	13.5	10.3	24.5	10.4	**0.8**	8.3	24.0	**8.4**	1.2
	VFNet [32]	R50	13.9	25.8	12.6	12.7	10.7	25.1	10.6	**0.5**	8.3	24.6	**8.0**	0.9
Two-Stage	Faster R-CNN [27]*	R50	27.0	29.9	25.5	17.6	10.4	23.8	10.4	**0.6**	8.6	23.5	**8.3**	0.9
	RS R-CNN [22]*	R50	19.7	28.9	18.2	41.1	10.2	23.5	10.3	2.0	8.1	23.0	**8.2**	**1.5**
DETR-like	D-DETR [34]*	R50	12.7	27.1	12.1	12.8	9.6	23.5	10.1	**0.9**	7.7	23.1	**8.2**	1.3
	UP-DETR [5]	R50	34.4	35.2	32.6	34.7	10.0	22.6	10.3	1.5	8.2	22.2	**7.5**	**1.3**
	DINO [31]	R50	13.6	26.9	13.4	14.8	10.6	23.5	10.2	1.9	8.9	22.8	**8.7**	**1.1**
OVOD	GLIP [13]	Swin-T	13.0	25.3	12.3	25.4	9.2	22.4	9.4	**0.7**	7.7	21.8	**7.8**	1.1
	G. DINO [17]	Swin-T	13.8	27.5	13.9	16.0	8.9	21.9	9.0	1.8	7.7	21.3	**7.4**	**1.5**
SOTA	Co-DETR [35]	Swin-L	10.8	23.0	10.8	12.9	8.6	20.2	8.4	0.8	6.7	19.3	**6.8**	**0.4**
	EVA [6]	ViT(EVA)	17.4	21.2	15.7	9.0	8.6	20.3	8.4	0.7	7.1	19.9	**6.8**	**0.6**
	MoCaE [25]	N/A	10.6	21.4	9.5	13.4	8.9	20.4	8.6	2.6	7.3	19.9	**7.0**	**0.8**

14. Li, X., Wang, W., Wu, L., Chen, S., Hu, X., Li, J., Tang, J., Yang, J.: Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In: Advances in Neural Information Processing Systems (NeurIPS) (2020)
15. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **42**(2), 318–327 (2020)
16. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common Objects in Context. In: The European Conference on Computer Vision (ECCV) (2014)
17. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499 (2023)
18. Mukhoti, J., Kulharia, V., Sanyal, A., Golodetz, S., Torr, P., Dokania, P.: Calibrating deep neural networks using focal loss. In: Larochelle, H., Ranzato, M., Hadsell,

R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 15288–15299. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper/2020/file/aeb7b30ef1d024a76f21a1d40e30c302-Paper.pdf
19. Munir, M.A., Khan, M.H., Khan, S., Khan, F.S.: Bridging precision and confidence: A train-time loss for calibrating object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11474–11483 (June 2023)
20. Munir, M.A., Khan, M.H., Sarfraz, M., Ali, M.: Towards improving calibration in object detection under domain shift. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems. vol. 35, pp. 38706–38718. Curran Associates, Inc. (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/fcd812a51b8f8d05cfea22e3c9c4b369-Paper-Conference.pdf
21. Munir, M.A., Khan, S., Khan, M.H., Ali, M., Khan, F.: Cal-DETR: Calibrated detection transformer. In: Thirty-seventh Conference on Neural Information Processing Systems (2023), https://openreview.net/forum?id=4SkPTD6XNP
22. Oksuz, K., Cam, B.C., Akbas, E., Kalkan, S.: Rank & sort loss for object detection and instance segmentation. In: The International Conference on Computer Vision (ICCV) (2021)
23. Oksuz, K., Cam, B.C., Kalkan, S., Akbas, E.: One metric to measure them all: Localisation recall precision (lrp) for evaluating visual detection tasks. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2021)
24. Oksuz, K., Joy, T., Dokania, P.K.: Towards building self-aware object detectors via reliable uncertainty quantification and calibration. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
25. Oksuz, K., Kuzucu, S., Joy, T., Dokania, P.K.: Mocae: Mixture of calibrated experts significantly improves object detection. arXiv preprint arXiv:2309.14976 (2023)
26. Popordanoska, T., Tiulpin, A., Blaschko, M.B.: Beyond classification: Definition and density-based estimation of calibration in object detection. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 585–594 (January 2024)
27. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **39**(6), 1137–1149 (2017)
28. Sakaridis, C., Dai, D., Van Gool, L.: Semantic foggy scene understanding with synthetic data. International Journal of Computer Vision **126**(9), 973–992 (Sep 2018), https://doi.org/10.1007/s11263-018-1072-8
29. Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., Sun, J.: Objects365: A large-scale, high-quality dataset for object detection. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
30. Wang, J., Zhang, W., Zang, Y., Cao, Y., Pang, J., Gong, T., Chen, K., Liu, Z., Loy, C.C., Lin, D.: Seesaw loss for long-tailed instance segmentation. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 9690–9699 (2020)
31. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y.: Dino: Detr with improved denoising anchor boxes for end-to-end object detection. arXiv preprint arXiv:2203.03605 (2022)
32. Zhang, H., Wang, Y., Dayoub, F., Sünderhauf, N.: Varifocalnet: An iou-aware dense object detector. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)

33. Zhang, S., Chi, C., Yao, Y., Lei, Z., Li, S.Z.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
34. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable {detr}: Deformable transformers for end-to-end object detection. In: International Conference on Learning Representations (ICLR) (2021)
35. Zong, Z., Song, G., Liu, Y.: Detrs with collaborative hybrid assignments training. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2023)