Calibrating the Full Predictive Class Distribution of 3D Object Detectors for Autonomous Driving

Cornelius Schröder[1], Marius-Raphael Schlüter and Markus Lienkamp[1]

***Abstract*— In autonomous systems, precise object detection and uncertainty estimation are critical for self-aware and safe operation. This work addresses confidence calibration for the classification task of 3D object detectors. We argue that it is necessary to regard the calibration of the full predictive confidence distribution over all classes and deduce a metric which captures the calibration of dominant and secondary class predictions. We propose two auxiliary regularizing loss terms which introduce either calibration of the dominant prediction or the full prediction vector as a training goal. We evaluate a range of post-hoc and train-time methods for CenterPoint, PillarNet and DSVT-Pillar and find that combining our loss term, which regularizes for calibration of the full class prediction, and isotonic regression lead to the best calibration of CenterPoint and PillarNet with respect to both dominant and secondary class predictions. We further find that DSVT-Pillar can not be jointly calibrated for dominant and secondary predictions using the same method.**

I. Introduction

Autonomous vehicles and other autonomous systems need to perceive their environment and detect and classify other traffic participants or objects in their vicinity. Based on these detections they plan their own behavior according to a high level goal. The planned behavior depends highly on the predicted classes and locations of the detected objects. Two attributes of object detectors are therefore important: It needs to precisely locate objects and classify them correctly, but it also must report meaningful information on the uncertainty of its predictions. If planning algorithms operate without precise uncertainty measures, they do not have the chance to adapt to the risk arising from the possibility of perceiving the environment incorrectly. Taking its own possible errors into account is a prerequisite for self-aware and explainable autonomous systems. In order to propagate uncertainties through all decision levels and incorporate them into planning, their initial estimates are of vital importance. We therefore aim to make the information about uncertainties arising in object detection more reliable. While there is uncertainty in both localization and classification, this research focuses on the classification uncertainty.

Even though current object detectors do usually not output their localization uncertainty, there are different methods for estimating it, such as Monte Carlo Dropout [1], [2], Deep Ensemble [3], Direct Modelling [4], [5] and Error Propagation [6]. However, not all of these approaches are feasible for real-world systems due to the computational overhead of Bayesian methods [7].

In contrast, a measure for the classification uncertainty is intrinsically available in all deep neural network based object detectors. They take an input $\mathbf{x} \in \mathcal{X}$ and output a confidence vector $\mathbf{p} \in [0,1]^C$ alongside the maximum likelihood estimates for the bounding box parameters. The confidence vector $\mathbf{p}$ can be interpreted as the discrete probability distribution over the C classes from set $\mathcal{C}$. As of today, usually only the maximum element p_i of $\mathbf{p}$ is regarded and determines i as the object's class if p_i exceeds a certain threshold. This disregard of the actual confidence and the secondary predictions is sub-optimal for safety critical autonomous systems. Simple examples can be constructed to show that the optimal behavior changes if the dominant class is predicted with a different confidence level or the confidence ranking of the secondary classes change: An object is classified as a vehicle with a high confidence requires a different planning than an object with high classification ambiguity between inanimate object or vulnerable road user. To assess the risk of its actions properly, a planning algorithm needs to have access to correct confidence estimates not only for the dominantly predicted class, but also for the less confident secondary class predictions.

Guo et al. [8] show that the confidence estimates of modern neural network classifiers are usually poorly calibrated, which means that the predicted classification confidence deviates from the observed empirical value. Küppers et al. [9] extend their analysis to 2D object detectors coming to a similar result. Perfect calibration can be expressed in two ways: The strong calibration condition

$$\mathbb{P}(Y = i \mid p_i(\mathbf{x})) = p_i, \text{ for all } i \in \mathcal{C}, \tag{1}$$

demands that for a set of detections the post-hoc measured empirical probability distribution of the object's ground truth class equals the predicted distribution including the non-dominant secondary classes [10]. The weak calibration condition

$$\mathbb{P}(Y = i \mid p_i(\mathbf{x})) = p_i, \text{ for } i = \operatorname{argmax}\ p_i(\mathbf{x}) \tag{2}$$

takes only the dominant class into account and allows a classifier to incorrectly rank secondary predictions while still regarding it as perfectly calibrated [8].

While there is a large body of research on the calibration of classifiers starting together with the early developments of machine learning and often stemming from meteorological

This research was funded by the Bavarian Research Foundation.

[1] Institute for Automotive Engineering,
Munich Institute of Robotics and Machine Intelligence,
School of Engineering and Design,
Technical University of Munich
cornelius.schroeder@tum.de

©2025 IEEE

or medical backgrounds [11]–[14], the extent of the miscalibration of modern neural networks is still disputed and might depend strongly on the specific architecture [8], [15]. Research about confidence calibration of object detectors is scarcer and mostly focused on 2D models operating on image data [9], [16]–[19]. Confidence calibration of 3D object detectors is performed by [20] and [21]. However, their works only employ the post-hocs methods Isotonic Regression and Temperature Scaling. To the best of our knowledge, a broad comparison of different classification calibration methods on state-of-the-art 3D object detectors for autonomous driving is not available.

Contribution

Our contribution is threefold:

- First, we deduce a new calibration metric called Full Detection Expected Calibration Error (Full D-ECE) from the strong calibration condition. In contrast to existing metrics, our metric therefore captures the calibration of the full predictive class distribution.
- Second, we introduce two different auxiliary loss terms $\mathcal{L}_{\text{DECE}}$ and $\mathcal{L}_{\text{FullDECE}}$, which induce regularization proportional to miscalibration during the training of object detectors and work as a train-time calibration method. $\mathcal{L}_{\text{FullDECE}}$ together with an adapted form of Isotonic Regression achieves best in class calibration for non transformer based 3D object detectors.
- Finally, we evaluate several combinations of train-time and post-hoc calibration methods on three different 3D object detectors on the Waymo Open dataset [22] for autonomous driving.

II. Related Work

The calibration of predictors is a long standing topic of research. Post-hoc methods are developed and used since several decades while train-time methods attracted interest only recently with increased application of deep neural networks.

A. Post-Hoc Calibration Methods

Post-hoc calibration methods aim to calibrate the confidence estimates of classifiers or object detectors once their training is finished. They can be divided into binning and scaling methods.

Binning methods achieve calibration by binning confidence values and remapping their estimates to the empirical accuracy or precision of the respective bin. Histogram Binning is a basic method which groups predictions according to their confidence levels, then assigns each detection the empirical accuracy calculated for the respective bin on the calibration data set [23]. Isotonic Regression extends Histogram Binning by optimizing both the confidence calibration within bins and the bin boundaries themselves [23]. Naeini et al. [24] introduce Bayesian Binning into Quantiles where they extend histogram binning by combining several binning models in a Bayesian probabilistic fashion. An Ensemble of Near Isotonic Regression [25] employs a similar strategy by finding a Bayesian combination of several models using Near Isotonic Regression [26], a variant of Isotonic Regression with relaxed monotonicity assumptions. Binning methods are non-parametric, meaning that they do not employ underlying assumptions about the data distribution.

Scaling methods do not manipulate the confidence estimates directly but rather rescale the logits before they are passed to the softmax or sigmoid activation function of the output layer. For binary classification problems or class-agnostic calibration, Platt Scaling [27] optimizes the parameters a and b of the function $y(x) = \sigma(ax + b)$, where y is the class confidence, x the logit before the activation function and σ the sigmoid function. During training of the classifiers the parameters are set to $a = 1$ and $b = 0$. In the subsequent calibration step, the parameters are fitted using a hold out dataset while the weights of the classifier remain fixed. Vector Scaling and Matrix Scaling [8] extend Platt Scaling for the class specific calibration of multi class classifiers. Instead of a scalar parameter, these methods use a vector or a matrix of learnable parameters to scale the logits for each class independently in case of Vector Scaling or dependently in case of Matrix Scaling. Temperature Scaling [8] is a simpler variant of Platt Scaling, which uses a single same parameter to scale all class logits. Despite its simplicity, Temperature Scaling is effective in calibrating the class confidences while at the same time preserving the accuracy of the trained classifier [28]. This makes Temperature Scaling a popular post-hoc calibration method.

In contrast to binning methods, scaling methods as parametric methods apply assumptions about the distribution of their parameters. This allows the parametric methods to optimize their parameters using less training data. While Platt Scaling and its descendants assume a normal distribution, methodically similar approaches such as Beta [29] and Dirichlet calibration [30] assume a Beta distribution for binary or a Dirichlet distribution for multi-class classification. Zhang et al. [28] show that combining parametric and non-parametric approaches can yield good calibration results.

B. Train-Time Calibration Methods

Label Smoothing, although designed to increase model performance and reduce overfitting, leads to inherently better calibrated models. It achieves it positive effect on calibration by removing the incentive for the model to place all probability mass on a single class while still enforcing separation in the logit values attributed to different classes [31], [32]. Similarly, Focal Loss [16] aims to mitigate the problem of hard and easy classes but also increases the entropy of the predictive confidence distribution. The resulting distribution is less pronounced and better calibrated [33]. Adaptive Focal Loss (AdaFocal) [34] is an extension of focal loss specifically designed for model calibration. During training, the model's miscalibration is constantly observed and the Focal Loss parameter γ is adapted to increase or decrease the gradients and thereby counteracting under-confident or overconfident predictions.

Other researchers introduce auxiliary loss terms which can be added as a regularization to any loss function. The Multiclass Difference of Confidence and Accuracy (MDCA) calculates the average confidence and accuracy on the current mini-batch and adds their difference as a regularizing component to the loss [35]. The auxiliary Multiclass Confidence and Localization Calibration loss [17] introduces two additional loss terms based on classification confidence uncertainty and localization uncertainty. During training, Monte-Carlo dropout is used to estimate the uncertainties of confidence predictions and bounding box parameters. These estimates are compared to the predicted confidence and the overlap with the ground-truth bounding box.

III. Methods

A. Existing Metrics for Calibration

The most widespread metric to measure the miscalibration of classifiers is the Expected Calibration Error (ECE) [24]. The ECE evaluates the expected divergence from the weak calibration condition (2). It approximates

$$\mathbb{E}(\mathbb{P}(Y = i \mid p_i(\mathbf{x})) - p_i), \text{ for } i = \text{argmax } p_i(\mathbf{x}) \quad (3)$$

by dividing the confidence predictions into B equally-spaced bins $\mathcal{B}$, with n_b out of total N predictions falling into a bin b, and comparing the difference between each bin's accuracy (fraction of correct predictions) and average confidence (mean predicted probability). The ECE is then computed as the weighted average of these calibration errors across all bins,

$$\text{ECE} = \sum_{b=1}^{B} \frac{n_b}{N} \cdot |\text{acc}(b) - \text{conf}(b)|. \quad (4)$$

Intuitively, a classifier is perfectly calibrated if the mean predicted confidence equals the empirical probability of correct predictions for all confidence bins.

In object detection, True Negatives and therefore accuracy are not defined. Küppers et al. [9] introduce a metric which uses precision instead of accuracy while also refining the binning scheme. Their metric enables binning based on bounding box parameters (such as position or size) in addition to confidence bins. This allows better evaluation of location-dependent miscalibration. However, binning across these additional bounding box dimensions requires an exponentially larger number of predictions to calculate meaningful precision and averaged confidence values. They define the Detection Expected Calibration Error (D-ECE) as

$$\text{D-ECE} = \sum_{b=1}^{B_{\text{total}}} \frac{n_b}{N} \cdot |\text{prec}(b) - \text{conf}(b)|, \quad (5)$$

where $B_{\text{total}} = \prod_{k=1}^{K} B_k$, obtained by iterating over all K dimensions used for binning.

B. Full D-ECE Calibration Metric

Both ECE and D-ECE are based on the weak condition for confidence calibration (2) and take only the predicted class into account. This is a substantial shortcoming when the optimal action of an autonomous system also depends on secondary class predictions, as in the example of an ambiguous detection with vulnerable road users involved. To be able to measure the calibration regarding the distribution of predicted probabilities for all classes, we introduce the Full Detection Expected Calibration Error (Full D-ECE). In analogy to (3), it is derived from the strong calibration condition (1) to estimate

$$\mathbb{E}(\mathbb{P}(Y = i \mid p_i(\mathbf{x})) - p_i), \text{ for all } i \in \mathcal{C}. \quad (6)$$

To calculate the Full D-ECE, we follow the same logic as the ECE and D-ECE only that we take the full vector of predicted probabilities for all C classes. This means that each detected object generates one "true positive" confidence prediction $p_{c,\text{ true}}$ and $C - 1$ "false positive" confidence predictions. We replace the precision of bin b with the ratio of the number of confidence predictions for correct classes falling into confidence bin b (regardless whether the ground truth class is assigned the highest probability or not) and total predictions in b, $\frac{|\mathcal{Y}_{\text{tp, b}}|}{n_b}$. The average bin confidence $\text{conf}(b)$ is now calculated over all confidence estimates for correct as well as wrong classes in bin b and therefore becomes $\frac{\sum_{p \in \mathcal{Y}_b} p}{n_b}$. More precisely and after reduction of n_b,

$$\text{Full D-ECE} = \sum_{b=1}^{B_{\text{total}}} \frac{||\mathcal{Y}_{\text{tp, b}}| - \sum_{p \in \mathcal{Y}_b} p|}{N}, \quad (7)$$

with p being the elements of the C-dimensional confidence vector for one of the N total detections and sets $\mathcal{Y}_b = \{p_{c,n} | c \in \mathcal{C}, n \in \mathcal{N}, p \in \mathcal{B}_b\}$ and $\mathcal{Y}_{\text{tp, b}} = \{p_{c,n} | c = c_{\text{true}}, n \in \mathcal{N}, p \in \mathcal{B}_b\}$, where $\mathcal{C}$ and $\mathcal{N}$ denote the sets of all classes and detections.

The Full D-ECE metric comes with another advantage: While the D-ECE metric is calculated only from the confidence of the predicted class, low confidence values are underrepresented, making the metric imprecise where the classifier is already relatively uncertain. The Full D-ECE on the other hand increases the number of available confidence values and has a fixed ratio of "true positive" and "false positive" confidence predictions. This results in more densely populated low-confidence bins, allowing more accurate statements about calibration in this range.

C. D-ECE and Full D-ECE as Auxiliary Loss Functions

We propose to enforce confidence calibration by including the confidence metrics D-ECE and Full D-ECE as auxiliary loss functions in addition to the base classification loss

$$\mathcal{L}_{\text{class}} = \mathcal{L}_{\text{base}} + \alpha \mathcal{L}_{\text{aux}}, \quad (8)$$

with α as the weight determining hyperparamter of the auxiliary loss term and $\mathcal{L}_{\text{aux}} = \mathcal{L}_{\text{DECE}} = \text{D-ECE}$ or $\mathcal{L}_{\text{aux}} = \mathcal{L}_{\text{FullDECE}} = \text{Full D-ECE}$.

This idea is not new since [35] and [14] also introduce a regularizing term based on the error between average confidence and empirical accuracy (or precision for object detection). However, both works do not use confidence binning to calculate the deviation from perfect calibration. This means that overconfident and underconfident predictions can cancel each other out, resulting in strong miscalibration as measured by the confidence-binned calibration metrics, but weak corresponding regularization in the loss function. In contrast, our approach utilizes the confidence-binned deviation from perfect calibration and regularizes the network using the D-ECE or Full D-ECE directly as auxiliary loss term weighted with a hyperparameter α. Both [35] and [14] state that confidence binning leads to a non-differentiable loss function, which is true since it has non-continuities at the bin borders. However, in practice this is not a problem and there are many common loss functions, such as ReLU, which are not differentiable. Because the probability of a confidence value landing exactly on the bin boundary converges to zero in the continuous case and is vanishingly small in the quantized case, we can use the standard ruleset of the PyTorch library for dealing with locally not-differentiable functions [36].

IV. Experiments

For our analysis, we select three LiDAR point cloud detectors based on their real-world applicability to autonomous vehicles and their architectural diversity. CenterPoint [37], a former state-of-the-art two-stage detector, remains widely used as the default detector in Autoware's open-source autonomous driving stack. PillarNet [38] offers real-time detection in a single-stage architecture. DSVT [39] employs dynamic sparse window attention for point cloud feature encoding. We evaluate these detectors' miscalibration and compare both post-hoc and train-time calibration methods to assess our auxiliary calibration losses and identify the most effective confidence calibration strategy.

A. Experimental Setup

We use the OpenPCDet framework [40] to train CenterPoint and PillarNet for 60 epochs on the Waymo Open Dataset's [22] training split downsampled to 2 Hz, while DSVT is trained for 24 epochs on the original training dataset sampled with 10 Hz. Each training is conducted with the largest feasible batch size of 45. For each network, we train a baseline using the default training regimen with voxel based focal loss for classification. Additionally, we train each model using either a combination of the standard loss and Adaptive Focal Loss with γ according to [34] or add D-ECE or Full D-ECE auxiliary loss terms to the focal loss. The calibration evaluation necessary during training uses 15 confidence bins and is performed on the last three batches for the Adaptive Focal Loss and on the last two batches for $\mathcal{L}_{\text{DECE}}$ and $\mathcal{L}_{\text{FullDECE}}$. We explore hyperparameter values $\alpha = [0.5, 1, 2, 5, 10, 20]$ for our auxiliary losses and find $\alpha = 1$ to produce the best results.

We further calibrate all models with the post-hoc methods Temperature Scaling, Platt Scaling, and Isotonic Regression using the implementation by [41]. For this purpose, we split the test set into two equal parts for calibration and evaluation. We fit the parameters for Temperature Scaling and Platt Scaling only on the dominant predictions (the classical approach aiming for the weak calibration condition (2)). Since Isotonic Regression has more degrees of freedom, we fit it not only on the dominant prediction but also incorporate the secondary predictions. Post-hoc calibration and evaluation of miscalibration uses 25 confidence bins.

B. Miscalibration of 3D Object Detectors

Tab. I provides a comparison of calibration performance for the baseline object detectors. In agreement with [8], we find that the more complex DSVT-Pillar network is less calibrated than the simpler CenterPoint and PillarNet networks ass measured by D-ECE. Nevertheless, it achieves a slightly better Full D-ECE, indicating improved calibration for secondary predictions. We note however that small confidence estimates are overrepresented in the Full D-ECE score because secondary predictions tend to be less confident while outweighing the dominant predictions by numbers if there are more than two classes.

TABLE I: Comparison of baseline object detectors

Network	mAP (L1)	mAP (L2)	D-ECE	Full D-ECE
CenterPoint [37]	0.734	0.673	0.159	0.075
PillarNet [38]	0.722	0.661	0.156	0.076
DSVT [39]	0.795	0.732	0.286	0.070

Fig. 1 shows the miscalibration of the baseline object detectors over the range of predicted confidence values taking into account either the dominant class prediction or the full confidence vector. We note that for DSVT-Pillar overconfidence is reduced for low to mid confidence levels when regarding calibration according to the strong condition (1). In contrast, CenterPoint and Pillarnet increase their overconfidence when calibration is evaluated on dominant and secondary predictions.

C. Confidence Calibration

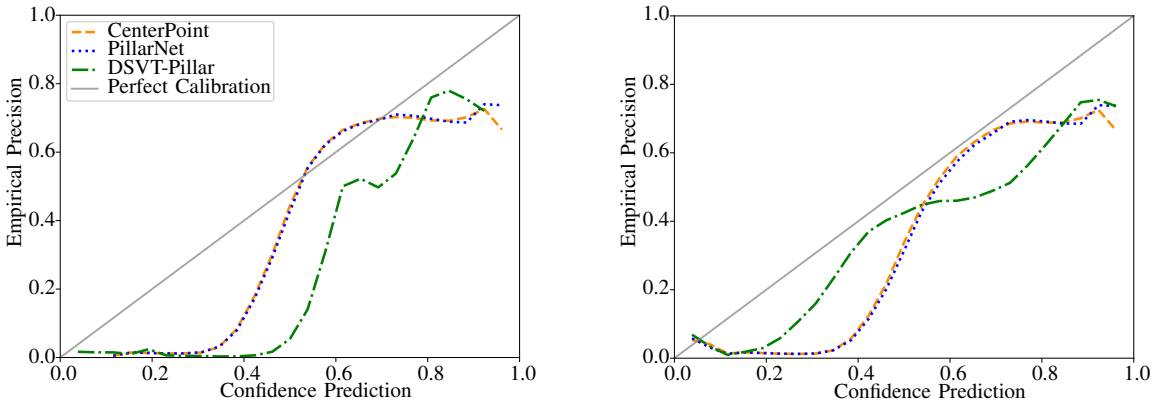
Tab. II summarizes the results of applying train-time and post-hoc calibration methods individually and in combination to the three baseline object detectors. Notably, each network achieves its highest mAP and mAPH scores either in its uncalibrated baseline form or when calibrated with Temperature or Platt Scaling.

Platt Scaling consistently improves calibration, as measured by both D-ECE and Full D-ECE, for all three baseline detectors and their variants with train-time calibration while preserving detection quality.

Isotonic Regression applied only to the dominant predictions, following [20] and [21], effectively reduces D-ECE across all model and train-time configurations. However, it slightly increases Full D-ECE and significantly reduces mAP and mAPH. When Isotonic Regression is instead fitted to

TABLE II: Calibration results: Train-time and posth-hoc methods. **Bold** marks the best results for each object detector.

Network	Calibration Method	mAP / mAPH (L1) ↑	mAP / mAPH (L2) ↑	D-ECE ↓	Full D-ECE ↓
CenterPoint [37]	uncalibrated	**0.734 / 0.708**	**0.673 / 0.649**	0.159	0.075
	Temperature Scaling	0.733 / 0.707	0.673 / 0.648	0.145	0.056
	Platt Scaling	**0.734 / 0.708**	**0.673 / 0.649**	0.080	0.032
	Isotonic Regression	0.696 / 0.666	0.640 / 0.611	0.078	0.106
	Isotonic Regression, full	0.693 / 0.664	0.633 / 0.606	0.031	**0.013**
	AdaFocal	0.676 / 0.649	0.612 / 0.588	↑ 0.355	0.135
	AdaFocal + Temperature Scaling	0.658 / 0.628	0.595 / 0.569	0.352	↑ 0.450
	AdaFocal + Platt Scaling	0.666 / 0.639	0.603 / 0.578	0.082	0.020
	AdaFocal + Isotonic Regression	↓ 0.618 / 0.587	↓ 0.558 / 0.530	0.063	0.085
	AdaFocal + Isotonic Regression, full	0.672 / 0.644	0.607 / 0.582	0.033	0.072
	$\mathcal{L}_{\text{DECE}}$	0.731 / 0.704	0.671 / 0.646	0.113	0.069
	$\mathcal{L}_{\text{DECE}}$ + Temperature Scaling	0.732 / 0.704	0.671 / 0.646	0.102	0.052
	$\mathcal{L}_{\text{DECE}}$ + Platt Scaling	0.732 / 0.705	0.671 / 0.646	0.078	0.044
	$\mathcal{L}_{\text{DECE}}$ + Isotonic Regression	0.690 / 0.658	0.633 / 0.604	0.075	0.109
	$\mathcal{L}_{\text{DECE}}$ + Isotonic Regression, full	0.706 / 0.675	0.648 / 0.619	0.040	0.017
	$\mathcal{L}_{\text{FullDECE}}$	0.731 / 0.705	0.669 / 0.645	0.167	0.080
	$\mathcal{L}_{\text{FullDECE}}$ + Temperature Scaling	0.729 / 0.703	0.668 / 0.644	0.139	0.054
	$\mathcal{L}_{\text{FullDECE}}$ + Platt Scaling	0.730 / 0.705	0.669 / 0.645	0.079	0.034
	$\mathcal{L}_{\text{FullDECE}}$ + Isotonic Regression	0.698 / 0.668	0.639 / 0.611	0.074	0.103
	$\mathcal{L}_{\text{FullDECE}}$ + Isotonic Regression, full	0.707 / 0.678	0.647 / 0.620	**0.030**	0.014
PillarNet [38]	uncalibrated	**0.722 / 0.687**	**0.661 / 0.629**	0.156	0.076
	Temperature Scaling	0.721 / 0.687	**0.661 / 0.629**	0.146	0.059
	Platt Scaling	**0.722 / 0.687**	**0.661 / 0.629**	0.081	0.031
	Isotonic Regression	0.688 / 0.649	0.631 / 0.595	0.073	0.103
	Isotonic Regression, full	0.694 / 0.655	0.637 / 0.601	0.031	**0.013**
	AdaFocal	0.627 / 0.593	0.565 / 0.535	↑ 0.376	↑ 0.143
	AdaFocal + Temperature Scaling	0.574 / 0.535	0.518 / 0.483	0.262	0.085
	AdaFocal + Platt Scaling	0.627 / 0.593	0.565 / 0.535	0.062	**0.013**
	AdaFocal + Isotonic Regression	0.586 / 0.549	0.529 / 0.496	0.056	0.082
	AdaFocal + Isotonic Regression, full	0.618 / 0.583	0.557 / 0.525	0.038	0.068
	$\mathcal{L}_{\text{DECE}}$	0.719 / 0.685	0.658 / 0.627	0.164	0.081
	$\mathcal{L}_{\text{DECE}}$ + Temperature Scaling	0.718 / 0.684	0.657 / 0.626	0.135	0.056
	$\mathcal{L}_{\text{DECE}}$ + Platt Scaling	0.719 / 0.685	0.658 / 0.627	0.077	0.035
	$\mathcal{L}_{\text{DECE}}$ + Isotonic Regression	0.685 / 0.646	0.627 / 0.592	0.076	0.107
	$\mathcal{L}_{\text{DECE}}$ + Isotonic Regression, full	0.695 / 0.657	0.636 / 0.602	**0.029**	0.014
	$\mathcal{L}_{\text{FullDECE}}$	0.718 / 0.684	0.657 / 0.626	0.169	0.079
	$\mathcal{L}_{\text{FullDECE}}$ + Temperature Scaling	0.717 / 0.682	0.656 / 0.624	0.144	0.056
	$\mathcal{L}_{\text{FullDECE}}$ + Platt Scaling	0.718 / 0.684	0.657 / 0.626	0.080	0.031
	$\mathcal{L}_{\text{FullDECE}}$ + Isotonic Regression	0.685 / 0.646	0.628 / 0.592	0.070	0.102
	$\mathcal{L}_{\text{FullDECE}}$ + Isotonic Regression, full	0.693 / 0.655	0.634 / 0.599	**0.029**	**0.013**
DSVT-Pillar [39]	uncalibrated	**0.795 / 0.770**	**0.732 / 0.709**	0.286	0.070
	Temperature Scaling	**0.795 / 0.770**	**0.732 / 0.709**	0.291	0.082
	Platt Scaling	**0.795 / 0.770**	**0.732 / 0.709**	0.076	0.064
	Isotonic Regression	0.773 / 0.744	0.713 / 0.686	0.047	0.092
	Isotonic Regression, full	0.770 / 0.746	0.708 / 0.685	0.233	0.025
	AdaFocal	0.792 / 0.768	0.730 / 0.707	↑ 0.324	0.088
	AdaFocal + Temperature Scaling	0.792 / 0.767	0.729 / 0.706	0.306	↑ 0.115
	AdaFocal + Platt Scaling	0.793 / 0.769	0.730 / 0.707	0.079	0.054
	AdaFocal + Isotonic Regression	0.767 / 0.739	0.707 / 0.681	**0.044**	0.080
	AdaFocal + Isotonic Regression, full	0.788 / 0.763	0.725 / 0.702	0.213	**0.021**
	$\mathcal{L}_{\text{DECE}}$	0.793 / 0.768	0.731 / 0.707	0.294	0.074
	$\mathcal{L}_{\text{DECE}}$ + Temperature Scaling	0.793 / 0.768	0.731 / 0.707	0.292	0.090
	$\mathcal{L}_{\text{DECE}}$ + Platt Scaling	0.794 / 0.769	0.731 / 0.707	0.078	0.064
	$\mathcal{L}_{\text{DECE}}$ + Isotonic Regression	↓ 0.766 / 0.736	↓ 0.707 / 0.679	0.046	0.094
	$\mathcal{L}_{\text{DECE}}$ + Isotonic Regression, full	0.777 / 0.752	0.709 / 0.686	0.234	0.026
	$\mathcal{L}_{\text{FullDECE}}$	0.792 / 0.766	0.730 / 0.706	0.294	0.073
	$\mathcal{L}_{\text{FullDECE}}$ + Temperature Scaling	0.792 / 0.766	0.730 / 0.706	0.295	0.088
	$\mathcal{L}_{\text{FullDECE}}$ + Platt Scaling	0.792 / 0.767	0.730 / 0.706	0.079	0.064
	$\mathcal{L}_{\text{FullDECE}}$ + Isotonic Regression	0.768 / 0.738	0.708 / 0.680	0.045	0.093
	$\mathcal{L}_{\text{FullDECE}}$ + Isotonic Regression, full	0.774 / 0.748	0.712 / 0.687	0.238	0.026



(a) Dominant predictions

(b) Dominant and secondary predictions

Fig. 1: When evaluating the miscalibration of the baseline detectors for dominant predictions only (a), CenterPoint and PointPillar are better calibrated than DSVT-Pillar. If the full vector of confidence predictions is regarded (b), DSVT-Pillar becomes less overconfident for low to mid confidence levels while the other detectors increase their overconfidence.

the full prediction vectors, the Full D-ECE score decreases substantially in all networks, regardless of whether they employ an auxiliary train-time method. For CenterPoint and PillarNet, the D-ECE also decreases. In contrast, DSVT-Pillar is not effectively calibrated in terms of D-ECE by this approach. This is due to the fact that the discrepancy in calibration between dominant predictions and the full confidence vector is larger than for the other detectors. As depicted in fig. 1, the full confidence vectors of DSVT suffer from less miscalibration than the dominant predictions alone. Therefore, using secondary predictions for isotonic regression does not scale the dominant predictions sufficiently, especially in the low confidence range where the number of secondary predictions outweighs dominant predictions.

Nevertheless, fitting Isotonic Regression to non-dominant predictions yields higher mAP for all networks and training configurations compared to using dominant predictions alone. The other post-hoc methods do not benefit from fitting on secondary predictions. Since they are restricted to a specific functional form, they easily overfit to the many low confidence secondary predictions. Isotonic Regression on the other hand, is only restricted to a monotonically increasing function with individual mapping for each of its bins. Therefore, it calibrates the fewer, usually dominant, predictions in the high-confidence bins unaffected by the large amount of low-confidence predictions.

In experiments with Adaptive Focal Loss, CenterPoint and PillarNet experience a pronounced drop in mAP and mAPH relative to baseline focal loss and our auxiliary train-time losses. This effect is not observed in DSVT-Pillar. The calibration results for Adaptive Focal Loss vary widely: as a stand-alone method, it worsens miscalibration in all detectors. However, when combined with Platt Scaling, it achieves decent calibration in D-ECE and Full D-ECE scores for each network.

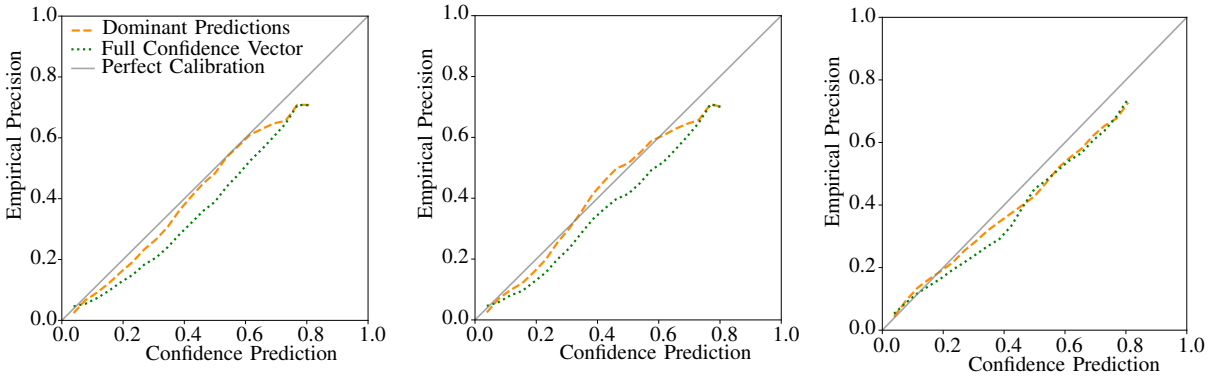
Our auxiliary training losses, $\mathcal{L}_{\text{DECE}}$ and $\mathcal{L}_{\text{FullDECE}}$, maintain detection quality on par with the baseline networks. $\mathcal{L}_{\text{DECE}}$ improves the calibration of CenterPoint relative to its uncalibrated baseline, but slightly degrades calibration in PillarNet and DSVT. Meanwhile, $\mathcal{L}_{\text{FullDECE}}$ is ineffective as a stand-alone calibration method; in combination with Isotonic Regression (standard or full), however, it achieves best in class calibration for D-ECE on CenterPoint and for both metrics on PillarNet.

We achieve the best calibration as measured by D-ECE for CenterPoint and PillarNet with the combination of $\mathcal{L}_{\text{FullDECE}}$ and Isotonic Regression fitted on the full confidence vector (fig. reffig:results). Interestingly, we are able to calibrate the dominant and secondary predictions simultaneously. Although the Full D-ECE scores for both networks are smaller than the D-ECE, fig. 2(a) and 2(b) suggest better calibration of the dominant class prediction. Further analysis shows, that the Full D-ECE metric is heavily influenced by the large amount of samples in the lowest confidence bin, where miscalibration is low.

In case of DSVT, we achieve effective calibration for both dominant and secondary confidence predictions using Adaptive Focal Loss and Isotonic Regression, as depicted in fig. 2(c). However, no combination of calibration methods was able to do so simultaneously. The decision on which confidence predictions to fit Isotonic Regression influences the calibration of dominant and secondary predictions differently since dominant predictions and full confidence vectors exhibit a different miscalibration pattern. It is therefore beneficial to choose the post-hoc method according to whether only dominant or all confidence predictions will be used in an application.

V. Conclusion

From our experiments we find that it is indeed possible to achieve good calibration of 3D object detectors not only for dominant but also for secondary class predictions. This



(a) CenterPoint, $\mathcal{L}_{\text{FullDECE}}$ + Isotonic Regression (full)

(b) PillarNet, $\mathcal{L}_{\text{FullDECE}}$ + Isotonic Regression (full)

(c) DSVT-Pillar, Adaptive Focal Loss + Isotonic Regression (dominant/full)

Fig. 2: We are able to simultaneously calibrate CenterPoint (a) and PillarNet (b) effectively for dominant predictions only and the full confidence vector using a combination of our training loss $\mathcal{L}_{\text{FullDECE}}$ and Isotonic Regression fitted on all predictions. Because simultaneous calibration of dominant predictions and full confidence vector does not succeed in case of DSVT-Pillar (c), we fit Isotonic Regression for calibration of dominant predictions and the full confidence vector only on dominant or all predictions.

distinction is especially important in safety critical applications. However, no single strategy works equally well for all detectors. If retaining the original detection quality is paramount, Platt Scaling is a good choice; either as standalone method or in combination with $\mathcal{L}_{\text{DECE}}$ or $\mathcal{L}_{\text{FullDECE}}$ auxiliary losses for Centerpoint and Pillarnet or combined with Adaptive Focal Loss for the more complicated DSVT-Pillar detector. For best-in-class calibration, CenterPoint and PillarNet benefit most from combining our regularizing loss term $\mathcal{L}_{\text{FullDECE}}$ with Isotonic Regression fitted on the entire confidence vector. The DSVT-Pillar transformer-based detector, however, reveals a different miscalibration pattern which makes it challenging to optimize both dominant and secondary predictions simultaneously. The best combination is Adaptive Focal Loss with Isotonic Regression; but it is important to decide before calibration whether the focus lies on dominant predictions or the full confidence vector. Future research should thus emphasize calibration for transformer-based networks. We furthermore encourage to evaluate next-generation 3D detectors not only on detection metrics but also on calibration quality and adaptability to various calibration strategies.

Acknowledgment

Author contributions: C. Schröder, as the first author, devised the essential concepts of the proposed metric and auxiliary loss terms, conducted the analysis and wrote the article. M.-R. Schlüter implemented the concepts in code. M. Lienkamp made an essential contribution to the concept of the research project. He revised the paper critically for important intellectual content. M. Lienkamp gives final approval for the version to be published and agrees to all aspects of the work. As a guarantor, he accepts responsibility for the overall integrity of the paper.

References

[1] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Proceedings of The 33rd International Conference on Machine Learning*, vol. 48, Jun 2016, pp. 1050–1059.

[2] Y. Gal, J. Hron, and A. Kendall, "Concrete dropout," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[3] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[4] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[5] J. Gast and S. Roth, "Lightweight probabilistic deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[6] J. Postels, F. Ferroni, H. Coskun, N. Navab, and F. Tombari, "Sampling-free epistemic uncertainty estimation using approximated variance propagation," in *Proceedings - 2019 International Conference on Computer Vision, ICCV 2019*, Oct. 2019, pp. 2931–2940.

[7] D. Feng, A. Harakeh, S. L. Waslander, and K. Dietmayer, "A review and comparative study on probabilistic object detection in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 9961–9980, 2022.

[8] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger, "On calibration of modern neural networks," *International Conference on Machine Learning*, pp. 1321–1330, 2017.

[9] Fabian Küppers, Jan Kronenberger, Amirhossein Shantia, and Anselm Haselhoff, "Multivariate confidence calibration for object detection," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020.

[10] D. Widmann, F. Lindsten, and D. Zachariah, "Calibration tests in multi-class classification: A unifying framework," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[11] M. H. DeGroot and S. E. Fienberg, "The comparison and evaluation of forecasters," *The Statistician*, vol. 32, no. 1/2, p. 12, 1983.

[12] J. S. Denker and Y. LeCun, "Transforming neural-net output levels to probability distributions," in *Advances in Neural Information Processing Systems 3, [NIPS Conference, Denver, Colorado, USA, November 26-29, 1990]*, 1990, pp. 853–859.

[13] X. Jiang, M. Osl, J. Kim, and L. Ohno-Machado, "Calibrating predictive model estimates to support personalized medicine," *Journal of the American Medical Informatics Association : JAMIA*, vol. 19, no. 2, pp. 263–274, 2012.

[14] G. Liang, Y. Zhang, X. Wang, and N. Jacobs, "Improved trainable calibration method for neural networks." in *BMVC*, 2020.

[15] M. Minderer, J. Djolonga, R. Romijnders, F. Hubis, X. Zhai, N. Houlsby, D. Tran, and M. Lucic, "Revisiting the calibration of modern neural networks," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 15 682–15 694.

[16] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318–327, 2020.

[17] B. Pathiraja, M. Gunawardhana, and M. H. Khan, "Multiclass confidence and localization calibration for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19 734–19 743.

[18] F. Küppers, J. Kronenberger, J. Schneider, and A. Haselhoff, "Bayesian confidence calibration for epistemic uncertainty modelling," in *2021 IEEE Intelligent Vehicles Symposium (IV)*, 2021, pp. 466–472.

[19] F. Küppers, A. Haselhoff, J. Kronenberger, and J. Schneider, *Confidence Calibration for Object Detection and Segmentation*, 2022, pp. 225–250.

[20] Di Feng, L. Rosenbaum, C. Glaeser, F. Timm, and K. Dietmayer, "Can we trust you? on calibration of a probabilistic object detector for autonomous driving." [Online]. Available: https://arxiv.org/pdf/1909.12358.pdf

[21] Y. Kato and S. Kato, "A conditional confidence calibration method for 3d point cloud object detection," in *2022 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 6/4/2022 - 6/9/2022, pp. 1835–1844.

[22] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[23] B. Zadrozny and C. Elkan, "Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers," *International Conference on Machine Learning*, 2001.

[24] Mahdi Pakdaman Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," *AAAI Conference on Artificial Intelligence*, 2015.

[25] Mahdi Pakdaman Naeini and G. Cooper, "Binary classifier calibration using an ensemble of near isotonic regression models," *Industrial Conference on Data Mining*, 2015.

[26] R. Tibshirani and H. Höfling, "Nearly-isotonic regression," *Technometrics*, 2011.

[27] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Adv. Large Margin Classif.*, vol. 10, Jun 2000.

[28] Jize Zhang, Bhavya Kailkhura, and T. Yong-Jin Han, "Mix-n-match : Ensemble and compositional methods for uncertainty calibration in deep learning," *International Conference on Machine Learning*, pp. 11 117–11 128, 2020.

[29] M. Kull, T. Silva Filho, and P. Flach, "Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers," *Artificial Intelligence and Statistics*, pp. 623–631, 2017.

[30] M. Kull, M. Perello Nieto, M. Kängsepp, T. Silva Filho, H. Song, and P. Flach, "Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[31] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[32] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, "Regularizing neural networks by penalizing confident output distributions," in *Proc. ICLR Workshop*, 2017.

[33] J. Mukhoti, V. Kulharia, A. Sanyal, S. Golodetz, P. Torr, and P. Dokania, "Calibrating deep neural networks using focal loss," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 15 288–15 299.

[34] A. Ghosh, T. Schaaf, and M. Gormley, "Adafocal: Calibration-aware adaptive focal loss," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 1583–1595.

[35] R. Hebbalaguppe, J. Prakash, N. Madan, and C. Arora, "A stitch in time saves nine: A train-time regularizing loss for improved neural network calibration," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 16 060–16 069.

[36] "Autograd mechanics — pytorch 2.5 documentation," 1/26/2025. [Online]. Available: https://pytorch.org/docs/stable/notes/autograd.html

[37] T. Yin, X. Zhou, and P. Krahenbuhl, " Center-based 3D Object Detection and Tracking ," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2021, pp. 11 779–11 788.

[38] G. Shi, R. Li, and C. Ma, "Pillarnet: Real-time and high-performance pillar-based 3d object detection." Berlin, Heidelberg: Springer-Verlag, 2022.

[39] H. Wang, C. Shi, S. Shi, M. Lei, S. Wang, D. He, B. Schiele, and L. Wang, "Dsvt: Dynamic sparse voxel transformer with rotated sets," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 13 520–13 529.

[40] O. D. Team, "Openpcdet: An open-source toolbox for 3d object detection from point clouds," https://github.com/open-mmlab/OpenPCDet, 2020.

[41] S. Kuzucu, K. Oksuz, J. Sadeghi, and P. K. Dokania, "On calibration of object detectors: Pitfalls, evaluation and baselines," in *Computer Vision – ECCV 2024*, pp. 185–204.