

LMD: Light-weight Prediction Quality Estimation for Object Detection in Lidar Point Clouds

Tobias Riedlinger^{1*}[0000-0002-1953-8607], Marius
Schubert^{1*}[0000-0002-9410-8949], Sarina Penquitt^{1*}[0000-0001-6586-0931],
Jan-Marcel Kezmann^{1*}, Pascal Colling²[0000-0001-5599-1786], Karsten
Kahl¹[0000-0002-3510-3320], Lutz Roese-Koerner², Michael Arnold²,
Urs Zimmermann², Matthias Rottmann¹[0000-0003-3840-0184]

¹ University of Wuppertal, School of Mathematics and Natural Sciences, IZMD,
Germany {riedlinger,schubert,kkahl,roettmann}@math.uni-wuppertal.de,
jankezzmann@t-online.de

² Aptiv Services Deutschland GmbH
{pascal.colling,lutz.roese-koerner,michael.arnold,urs.zimmermann}
@aptiv.com

Abstract. Object detection on Lidar point cloud data is a promising technology for autonomous driving and robotics which has seen a significant rise in performance and accuracy during recent years. Particularly uncertainty estimation is a crucial component for down-stream tasks and deep neural networks remain error-prone even for predictions with high confidence. Previously proposed methods for quantifying prediction uncertainty tend to alter the training scheme of the detector or rely on prediction sampling which results in vastly increased inference time. In order to address these two issues, we propose LidarMetaDetect (LMD), a light-weight post-processing scheme for prediction quality estimation. Our method can easily be added to any pre-trained Lidar object detector without altering anything about the base model and is purely based on post-processing, therefore, only leading to a negligible computational overhead. Our experiments show a significant increase of statistical reliability in separating true from false predictions. We propose and evaluate an additional application of our method leading to the detection of annotation errors. Explicit samples and a conservative count of annotation error proposals indicates the viability of our method for large-scale datasets like KITTI and nuScenes. On the widely-used nuScenes test dataset, 43 out of the top 100 proposals of our method indicate, in fact, erroneous annotations.

Keywords: Lidar point cloud · object detection · uncertainty estimation · annotation quality.

* Equal contribution.

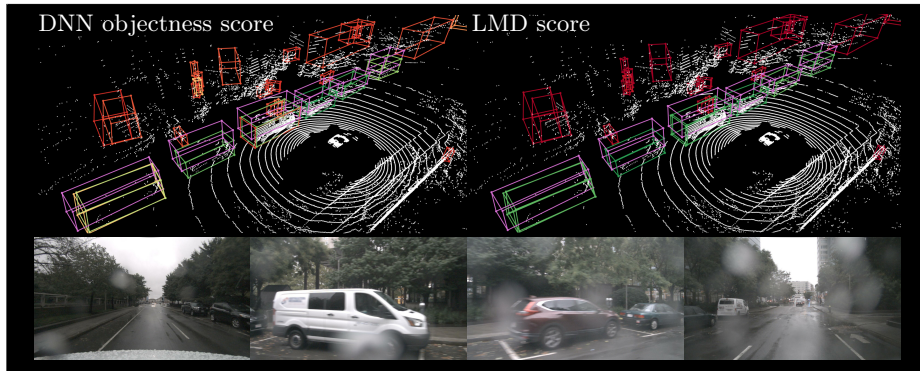


Fig. 1: Prediction of a Lidar point cloud object detector with the native objectness score (*left*) and LMD meta classifier scores (*right*) and corresponding camera images below. Ground truth annotations are depicted in pink while predictions are color-coded from red (low confidence) to green (high confidence). Detections based on the objectness score are highly threshold-dependent and may lead to false positive detections. Detections based on LMD scores are more reliable and separate true from false predictions more sharply.

1 Introduction

In recent years, deep learning has achieved great advances in the field of 3D object detection on Lidar data [9,21,22,24]. Deep neural network (DNN) architectures for this task are well-developed, however, there is little work in the area of uncertainty quantification (UQ) for such models [15,2,11,12,23]. UQ is crucial for deployment of DNN-based object detection in the real world, since DNNs as statistical models statistically make erroneous predictions. Down-stream algorithms are supposed to further process the predictions of perception algorithms and rely on statistically accurate and meaningful UQ. Aleatoric uncertainty is usually estimated by adding variance parameters to the network prediction and fitting them to data under a specific assumption for the distribution of residuals [2,12,11,5,4]. Such approaches usually alter the training objective of the detector by appealing to the negative log-likelihood loss for normally distributed residuals. Epistemic uncertainty is oftentimes estimated via Monte-Carlo (MC) dropout [2] or deep ensembles [23]. In such approaches, model sampling leads to a significant increase in inference time. Inspired by lines of research [19,16] in the field of 2D object detection on camera images, we develop a framework for UQ in 3D object detection for Lidar point clouds. This approach does not alter the training objective and can be applied to any pre-trained object detector and does not require prediction sampling. Our framework, called LidarMetaDetect (short LMD), performs two UQ tasks: (1) meta classification, which aims at estimating the probability of a given prediction being a true positive vs. being a false positive; (2) meta regression, which estimates the localization quality of a

prediction compared with the ground truth. Note that, outside the context of UQ for DNNs, the terms meta classification and meta regression refer to different concepts, see [10] and [20], respectively. LMD operates as a post-processing module and can be combined with any DNN without modifying it. Our methods learn on a small sample of data to assess the DNN’s reliability in a frequentist sense at runtime, i.e., in the absence of ground truth. In essence, we handcraft a number of uncertainty scores on bounding box level, by which we convert both UQ tasks into structured machine learning tasks. To the best of our knowledge, our method is the first purely post-processing-based UQ method for 3D object detection based on Lidar point clouds. We conduct in-depth numerical studies on the KITTI [6], nuScenes [1] as well as a propriety dataset. We include comparisons of our methods with baseline methods on common uncertainty quantification benchmarks, ablation studies of relevant parameters and the relevance of our uncertainty features. This is complemented with down stream tasks where (1) we demonstrate that our UQ increases the separation of true and false predictions and leads to well-calibrated confidence estimates and (2) we show that our UQ can be utilized for the detection of erroneous annotations in Lidar object detection datasets. We evaluate our method’s annotations error detection capabilities by reviewing its proposals on moderate samples from KITTI and nuScenes. Our contributions can be summarized as follows:

- We develop the first purely post-processing based UQ framework for 3D object detection in Lidar point clouds.
- We compare our UQ methods to baselines and show that they clearly outperform the DNN’s built-in estimates of reliability.
- We find annotation errors in the most commonly used publicly available Lidar object detection datasets, i.e., KITTI and nuScenes.

We make our code publicly available at <https://github.com/JanMarcelKezmann/MetaDetect3D>.

2 Related Work

In recent years, technologically sophisticated methods such as perception in Lidar point clouds have received attention in the UQ branch due to their potential industrial relevance in the autonomous driving sector. Methods for 3D object detection roughly fall into the categories of aleatoric and epistemic UQ. Aleatoric UQ methods usually build on estimating distributional noise by adding a variance output for each regression variable while epistemic UQ methods utilize some kind of model sampling either appealing to MC dropout or deep ensembles. Meyer et al. [11] estimate aleatoric uncertainty by a two-dimensional discretization scheme over the Lidar range and introducing a variance-weighted regression loss for a multi-modal distributional prediction in order to improve detection performance. Meyer and Thakurdesai [12] estimate aleatoric uncertainty by adding scale regression variables to the network output, modeling Laplace-distributed residuals under a label noise assumption via a Kullback-Leibler divergence loss. Feng et al. [5] estimate heteroscedastic aleatoric uncertainty for the

region proposal and the detection head of an object detector separately by modeling diagonal-covariance normally distributed bounding box regression. Feng et al. [4] achieve joint estimation of aleatoric and epistemic UQ by adding regression variables that model the covariance diagonal of a multi-variate normal distribution of the four bounding box parameters alongside MC dropout total variance for the epistemic component. Chen et al. [2] extract aleatoric uncertainty information from a self-supervised projection-reconstruction mechanism propagated to 3D object detection on camera images. Further, epistemic uncertainty of object localization is quantified via MC dropout. Yang et al. [23] perform UQ for 3D object detection on Lidar and extend the multi-input multi-output model MIMO [8] which modifies the network to be supplied simultaneously with n inputs and providing n outputs. This simulates a deep ensemble at inference time at the cost of increased memory consumption for input and output layers.

In the field of 2D object detection in camera images by DNNs, methods for UQ have been developed in a series of works [19,16] related with research on UQ in semantic segmentation [18,17]. Schubert et al. [19] utilize the pre-NMS anchor statistics in a post-processing approach to obtain box-wise confidence and *IoU*-estimates. Riedlinger et al. [16] use instance-wise gradient scores in a post-processing scheme to obtain calibrated uncertainty estimates improving detection performance. Inspired by these lines of research, we develop a framework for UQ in 3D object detection for Lidar point clouds. We use lightweight post-processing models on top of a pre-trained Lidar point cloud object detector in order to obtain improved uncertainty and *IoU*-estimates. In contrast to previous work, our approach has the advantage that it may be applied to any pre-trained object detector without alteration of training or architecture and does not carry the computational and memory cost of sampling weights in a Bayesian manner like MC dropout or deep ensembles. We show that this approach leads to more reliable object detection predictions and that it can be applied in an intuitive way in order to detect annotation errors in object detection datasets.

3 Proposed Method

In this section we describe our post-processing mechanism and how it can be applied to improve detection performance and to detect annotation errors. Our method assumes an object detector $f(\cdot)$ which maps point clouds \mathbf{X} to a list of N bounding boxes

$$f(\mathbf{X}) = \{\hat{b}^1, \dots, \hat{b}^N\}. \quad (1)$$

Point clouds $\mathbf{X} = (\mathbf{p}_1, \dots, \mathbf{p}_{N_{\text{pt}}})$ consist of Lidar points $\mathbf{p} = (x, y, z, r) \in \mathbb{R}^4$ represented by three coordinates (x, y, z) and a reflectance value r each. Bounding boxes are represented by features $\hat{b}^j(\mathbf{X}) = (\hat{x}^j, \hat{y}^j, \hat{z}^j, \hat{\ell}^j, \hat{w}^j, \hat{h}^j, \hat{\theta}^j, \hat{s}^j, \hat{\pi}_1^j, \dots, \hat{\pi}_C^j)$. Here, $\hat{x}^j, \hat{y}^j, \hat{z}^j, \hat{\ell}^j, \hat{w}^j, \hat{h}^j, \hat{\theta}^j$ define the bounding box geometry, \hat{s}^j is the objectness score and $(\hat{\pi}_1^j, \dots, \hat{\pi}_C^j)$ is the predicted categorical probability distribution. The latter defines the predicted class $\hat{\kappa}^j = \operatorname{argmax}_{c=1, \dots, C} \hat{\pi}_c^j$ while the objectness score \hat{s}^j is the model’s native confidence estimate for each prediction. Out

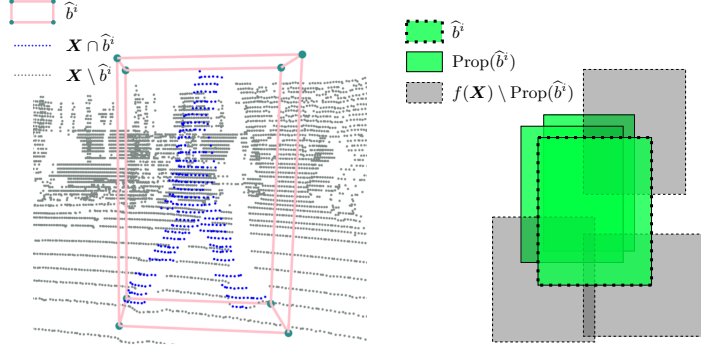


Fig. 2: Left: Illustration of the P^i and Φ^i features counting Lidar points falling into a given predicted box. From the points $\mathbf{X} \cap \hat{b}^i$, reflection statistics are generated. Right: Schematic illustration of the proposal set $\text{Prop}(\hat{b}^i)$ for a given predicted box \hat{b}^i (here, in two dimensions for simplicity). From the proposal boxes, further pre-NMS statistics are derived.

of the N bounding boxes, only a small amount N_{NMS} will be left after non-maximum suppression (NMS) filtering and contribute to the final prediction of the detector

$$\text{NMS}[f(\mathbf{X})] = \left\{ \hat{b}^i : i \in I_{\text{NMS}} \right\}, \quad (2)$$

where we let $I_{\text{NMS}} \subset \{1, \dots, N\}$ denote the post-NMS index set indicating survivor boxes.

LMD Features. From this information we generate geometrical and statistical features for each $\hat{b}^i \in \text{NMS}[f(\mathbf{X})]$ for the purpose of UQ. In addition to the bounding box features

$$\hat{\phi}^i := \{\hat{x}^i, \hat{y}^i, \hat{z}^i, \hat{\ell}^i, \hat{w}^i, \hat{h}^i, \hat{\theta}^i, \hat{s}^i, \hat{\kappa}^i\} \quad (3)$$

of \hat{b}^i we compute the geometric features *volume* $V^i = \hat{\ell}^i \hat{w}^i \hat{h}^i$, *surface area* $A^i = 2(\hat{\ell}^i \hat{w}^i + \hat{\ell}^i \hat{h}^i + \hat{w}^i \hat{h}^i)$, *relative size* $F^i = V^i/A^i$, *number of Lidar points* $P^i = |\mathbf{X} \cap \hat{b}^i|$ within \hat{b}^i and *fraction of Lidar points* $\Phi^i = P^i/|\mathbf{X}|$ in \hat{b}^i , see fig. 2 on the left for an illustration. Moreover, each Lidar point that falls into \hat{b}^i (i.e., in $\mathbf{X} \cap \hat{b}^i$) has a reflectance value r . We add the maximal (ρ_{max}^i), mean (ρ_{mean}^i) and standard deviation (ρ_{std}^i) over all reflectance values of points in \hat{b}^i . Lastly, for each \hat{b}^i , we take the pre-NMS statistics into consideration which involves all proposal boxes in $f(\mathbf{X})$ that are NMS-filtered by \hat{b}^i , i.e., the pre-image

$$\text{Prop}(\hat{b}^i) := \text{NMS}^{-1}[\{\hat{b}^i\}]. \quad (4)$$

These are characterized by having a significant three-dimensional IoU_{3D} with \hat{b}^i , see fig. 2 on the right. The number of proposal boxes $N^i := |\text{Prop}(\hat{b}^i)|$ is an

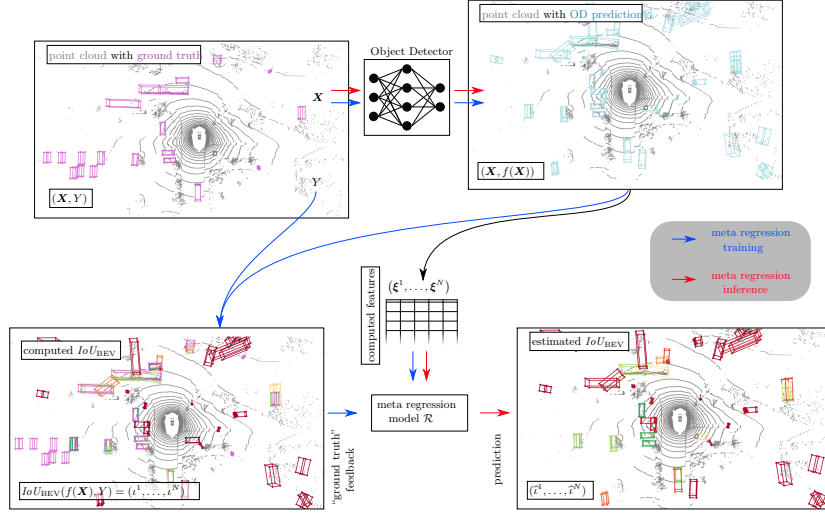


Fig. 3: Schematic illustration of the LMD meta regression pipeline. Training of the model is based on the output $f(\mathbf{X})$ of a fixed (frozen) object detector and the bounding box ground truth Y . Meta classification follows the same scheme with binary training targets $\tau^i = 1_{\{\iota^i > 0.5\}}$.

important statistics since regions with more proposals are more likely to contain a true prediction. We further derive minimum, maximum, mean and standard deviation statistics over proposal boxes $\hat{b} \in \text{Prop}(\hat{b}^i)$ for all

$$m^i \in \hat{\phi}^i \cup \{V^i, A^i, F^i, P^i, \Phi^i, \rho_{\max}^i, \rho_{\text{mean}}^i, \rho_{\text{std}}^i\}, \quad (5)$$

as well, as the IoU_{3D} and bird-eye intersection over union IoU_{BEV} values between \hat{b}^i and all proposals $\text{Prop}(\hat{b}^i)$. Overall, this amounts to a vector $\xi^i(\mathbf{X})$ of length $n = 90$ consisting of co-variables (features) on which post-processing models are fit in order to predict the IoU_{BEV} between \hat{b}^i and the ground truth or classify samples as true (TP) or false positives (FP). We call a box a TP if $IoU_{BEV} \geq 0.5$, otherwise we declare it FP.

Post-Processing. On an annotated hold-out dataset \mathcal{D}_{val} (consisting of point cloud-annotation tuples (\mathbf{X}, Y)), we compute a structured dataset denoted $\mathbf{X} = (\xi^1, \dots, \xi^{N_{\text{val}}}) \in \mathbb{R}^{n \times N_{\text{val}}}$ consisting of feature vectors for each of the N_{val} predicted boxes over all of \mathcal{D}_{val} . The illustration of our method in fig. 3 shows this scheme for one particular Lidar frame (\mathbf{X}, Y) and the respective prediction on it. Further, we compute $\iota^i := IoU_{BEV}(\hat{b}^i(\mathbf{X}), Y)$ between prediction and ground truth form \mathcal{D}_{val} as target variables $Y = (\iota^1, \dots, \iota^{N_{\text{val}}}) \in \mathbb{R}^{N_{\text{val}}}$. We then fit a light-weight (*meta*-) regression model $\mathcal{R} : \xi^i \mapsto Y_i$ on (\mathbf{X}, Y) which acts as post-processing module of the detector in order to produce IoU_{BEV} -estimates

$\hat{v}^i := \mathcal{R}(\xi^i)$ for each detection \hat{b}^i . Similarly, we fit a binary (*meta-*) *classification model* \mathcal{C} obtaining the binary targets $1_{\{Y>0.5\}}$ which allows us to generate alternative confidence estimates $\hat{\tau}^i := \mathcal{C}(\xi^i) \in [0, 1]$ for each prediction \hat{b}^i in post-processing. Note that \mathcal{C} is a potentially non-monotonous function of the features ξ^i and, therefore, can change the obtained confidence ranking per frame and influence detection performance as opposed to re-calibration methods [14,7].

Meta classification empirically turns out to produce confidence estimates which are both, sharper (in the sense of separating TPs from FPs) and better calibrated than those produced natively by the detector, i.e., the objectness score. However, when regarding the cases of disagreement between the computed IoU_{BEV} and \mathcal{C} , we frequently find that \mathcal{C} is to be trusted more than the computed IoU_{BEV} due to missing annotations. We use this observation in order to generate proposals (in descending estimation $\hat{\tau}^i$) based on the object detector in comparison with the given ground truth (FP according to the ground truth, i.e., $v^i < 0.5$) that serve as suggestions of annotation errors.

4 Numerical Results

In this section we study meta classification and meta regression performance for two benchmark datasets as well as a proprietary dataset by Aptiv. The meta classification results are presented in terms of accuracy and area under receiver operator characteristics curve (*AUROC* [3]) and the meta regression results are presented in terms of R^2 . We compare our uncertainty quantification method LidarMetaDetect (LMD) with two baseline methods (score, box features). Moreover, we detect annotation errors on both benchmark datasets using LMD.

Implementation Details. We implemented our method in the open source MMDetection3D toolbox [13]. For our experiments, we consider the PointPillars [9] and CenterPoint [24] architectures. The mean average precision ($mAP@IoU_{0.5}$) for KITTI based on IoU_{BEV} is 69.0 for CenterPoint and 68.8 for PointPillars. On KITTI, the $mAP@IoU_{0.5}$ based on IoU_{3D} is 64.2 for CenterPoint and 68.8 for PointPillars and for Aptiv, the $mAP@IoU_{0.5}$ based on IoU_{3D} is 39.5 for CenterPoint and 43.7 for PointPillars. NuScenes performance is given as a weighted sum of mAP as well as the nuScenes detection score (NDS). For CenterPoint, the mAP is 57.4 and the NDS is 65.2 and for PointPillars the mAP is 40.0 and the NDS is 53.3. For KITTI and Aptiv, the models were trained individually while available public model weights from MMDetection3D are used for nuScenes. The performance results obtained have all been evaluated on respective test datasets. For KITTI, the images and associated point clouds are split scene-wise, such that the training set consists of 3,712, the validation set of 1,997, and the test set of 1,772 frames. For nuScenes, the validation set is split scene-wise into 3,083 validation and 2,936 test frames. The Aptiv dataset consists of 50 sequences, split into 27, 14, 9 sequences with about 145K, 75K, 65K cuboid annotations for training, validation and testing, respectively. Every sequence is about two minutes long while every fifth point cloud is annotated. The covered locations

are countryside, highway and urban from and around (anonymous). The dataset includes four classes: 1. smaller vehicles likes cars and vans, 2. larger vehicles like busses and trucks, 3. pedestrians and 4. motorbikes and bicycles.

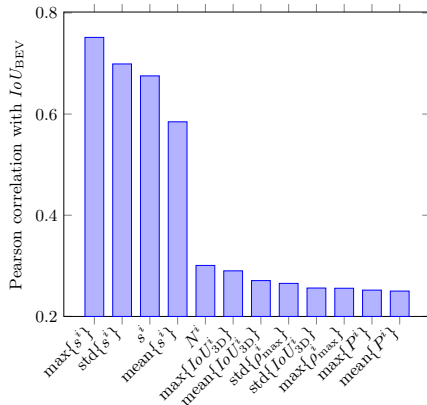


Fig. 4: Strongest correlation coefficients for constructed box-wise features and IoU_{BEV} for the CenterPoint architecture on the nuScenes test dataset and a score threshold $\tau = 0.1$.

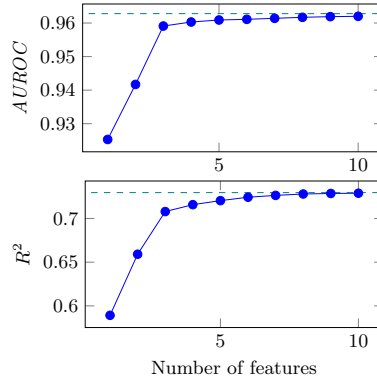


Fig. 5: Feature selection via greedy heuristic for CenterPoint, nuScenes and score threshold 0.1. *Top*: meta classification $AUROC$. *Bottom*: meta regression R^2 . The dashed line shows the performance when incorporating all features (LMD).

Correlation of Box-wise Features with the IoU_{BEV} . Figure 4 shows the Pearson correlation coefficients of the constructed box-wise dispersion measures with the IoU_{BEV} of prediction and ground truth for CenterPoint on the nuScenes test dataset. The score features have strong correlations (> 0.5) with the IoU_{BEV} . Note that, although the four score-related features show the highest individual correlation, these features may be partially redundant. The number of candidate boxes N^i is also reasonably correlated with the IoU_{BEV} (0.3007), whereas the remaining features only show a minor correlation (< 0.3). However, they may still contribute to higher combined explanatory information in meta classification.

Comparison of Different Meta Classifiers and Regressors. Different models can serve as post-processing modules for meta classification (\mathcal{C}) and meta regression (\mathcal{R} , see section 3). For meta classification, the meta models under consideration are logistic regression (LogReg), random forest (RF), gradient boosting (GB) and a multilayer perceptron (MLP) with two hidden layers. For meta regression, analogous regression models are used, only the logistic regression is replaced with a ridge regression (RR).

The respective meta models are trained on the box-wise features ξ^i of the validation sets \mathcal{D}_{val} and evaluated on the features of the test sets which are

Table 1: Comparison of meta classification accuracy and $AUROC$ as well as meta regression R^2 values for the score baseline, bounding box features and LMD for CenterPoint and nuScenes test dataset with score threshold 0.1; higher values are better. Bold numbers indicate the highest performance and underlined numbers represent the second highest (row-wise). Models used are Logistic Regression (LogReg), Ridge Regression (RR), Random Forest (RF), Gradient Boosting (GB) and a Multi Layer Perceptron (MLP).

Method	Meta Classification								Meta Regression			
	Accuracies				$AUROC$ s				R^2			
	LogReg	RF	GB	MLP	LogReg	RF	GB	MLP	RR	RF	GB	MLP
Score	0.8777	0.8524	0.8772	<u>0.8773</u>	0.8644	0.8617	0.8623	<u>0.8640</u>	0.4641	0.4675	<u>0.4733</u>	0.4751
Box Features	0.8877	<u>0.9049</u>	0.9203	0.8975	0.9056	<u>0.9454</u>	0.9529	0.9293	0.5292	<u>0.6681</u>	0.6792	0.6249
LMD	0.9118	0.9166	0.9297	<u>0.9200</u>	0.9450	<u>0.9581</u>	0.9628	0.9530	0.6451	<u>0.7242</u>	0.7296	0.7122

disjoint from \mathcal{D}_{val} . LMD uses all available features to train the meta models, whereas in the score baseline only the score of the prediction \hat{s}^i is used to fit the meta model. For the bounding box features baseline, the box features of the prediction $\hat{\phi}^i$ are used, in which the score \hat{s}^i is also included. Table 1 presents meta classification accuracy and $AUROC$ as well as meta regression R^2 for the CenterPoint architecture on the nuScenes dataset. For the score baseline, all meta models perform similarly well. For the meta classification accuracy there are differences of up to 2.53 percent points (pp), for the $AUROC$ of at most 0.27 pp and for meta regression R^2 of up to 1.10 pp. For the box features the maximum differences increase to 3.26 pp in terms of accuracy, to 4.73 pp for $AUROC$ and to 15.00 pp for R^2 . In particular, for the box features and LMD, the non-linear models (RF, GB, MLP) outperform the linear model in both learning tasks. LMD outperforms the baselines box features/score by 0.94/5.20 pp in terms of accuracy, by 0.99/9.84 pp in terms of $AUROC$ and by 5.04/25.45 pp in terms of R^2 . If overfitting of the meta model is made unlikely by choosing appropriate hyperparameters, the performance of the meta model typically benefits from adding more features, since the available information and number of parameters for fitting are increased. Overall, GB outperforms all other meta models, especially when multiple features are used to train and evaluate the respective learning task. Therefore, only results based on GB are shown in the following experiments. This finding may be attributed to the efficient fitting procedure and discontinuous nature of GB models.

Comparison for Different Datasets and Networks. Table 2 shows meta classification accuracy and $AUROC$ as well as meta regression R^2 for all network-dataset combinations based on GB models. In all cases LMD outperforms both baselines and the bounding box features outperform the score baseline. This is to be expected, since the score is contained in the box features and the box features are contained in the set of features of LMD. The improvement from the score baseline to LMD ranges from 0.83 to 6.76 pp in terms of meta classification accuracy,

Table 2: Comparison of meta classification accuracy and $AUROC$ as well as meta regression R^2 for the score baseline, bounding box features and LMD for all available network-dataset combinations with IoU_{BEV} threshold 0.5, score threshold 0.1 and GB as meta model. Bold numbers indicate the highest performance and underlined numbers represent the second highest (row-wise).

		Meta Classification						Meta Regression		
		Accuracies			$AUROC$ s			R^2		
Dataset	Network	Score	Box	LMD	Score	Box	LMD	Score	Box	LMD
KITTI	PointPillars	0.8921	<u>0.8931</u>	0.9004	0.9530	<u>0.9537</u>	0.9592	0.7108	<u>0.7131</u>	0.7287
	CenterPoint	0.8688	<u>0.8691</u>	0.8806	0.9274	<u>0.9343</u>	0.9466	0.6235	<u>0.6472</u>	0.6840
nuScenes	PointPillars	0.8398	<u>0.8708</u>	0.8915	0.8129	<u>0.9002</u>	0.9280	0.4055	<u>0.5593</u>	0.6413
	CenterPoint	0.8772	<u>0.9203</u>	0.9297	0.8623	<u>0.9529</u>	0.9628	0.4732	<u>0.6792</u>	0.7296
Aptiv	PointPillars	0.7939	<u>0.8489</u>	0.8615	0.8558	<u>0.9274</u>	0.9396	0.5096	<u>0.6568</u>	0.6924
	CenterPoint	0.8265	<u>0.8440</u>	0.8548	0.8914	<u>0.9134</u>	0.9275	0.5456	<u>0.6286</u>	0.6591

from 0.62 to 10.51 pp in terms of $AUROC$ and from 1.79 to 25.64 pp in terms of meta regression R^2 . The improvement from the bounding box features to LMD ranges from 0.73 to 2.07 pp in terms of meta classification accuracy, from 0.55 to 2.78 pp in terms of $AUROC$ and from 1.56 to 8.20 pp in terms of meta regression R^2 . This illustrates that the addition of features, other than just the bounding box features themselves, has a significant impact on meta classification and meta regression performances and, therefore, separation of TP and FP predictions.

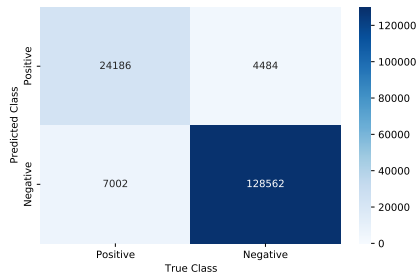


Fig. 6: Confusion matrix of a GB classifier for LMD on CenterPoint, nuScenes and score threshold 0.1.

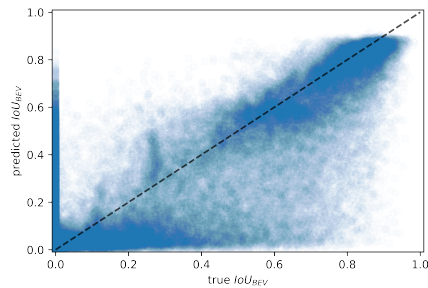


Fig. 7: Box-wise scatter plot of true IoU_{BEV} and predicted IoU_{BEV} values for LMD on CenterPoint, nuScenes and score threshold 0.1. The predictions are based on a GB regressor.

For CenterPoint and nuScenes, the confusion matrix fig. 6 shows that the GB classifier based on LMD identifies most TPs and true negatives. Therefore, predictions that are in fact FPs are also predicted as FPs. Note, that here we regard “meta” true negatives conditional on the detectors prediction (each ex-

ample is a detection TP or FP that is binarily classified). The values on the off-diagonals indicate the errors of the meta classifier. 7,002 predictions are predicted as FPs even though they are TPs. In contrast, 4,484 predictions are predicted as TPs, even though they are actually FPs. Figure 7 shows a scatter plot of the true IoU_{BEV} of prediction and ground truth and the IoU_{BEV} estimated by LMD meta regression based on a GB model, where each point represents one prediction. Well-concentrated points around the identity (dashed line) indicate well-calibrated IoU_{BEV} -estimates and, therefore, object-wise quality estimates. We observe a cluster of under-estimations for small IoU_{BEV} below the diagonal (bottom left) and false positives, vertically above true $IoU_{BEV} = 0$ on the left.

Feature Selection for Meta Classification and Meta Regression. Overall, LMD is based on 90 features, which partly describe very similar properties. In order to get a subset of features which contains as few redundancies as possible but is still powerful, we apply a greedy heuristic. Starting with an empty set, a single feature that improves the meta prediction performance maximally is added iteratively. Figure 5 shows results in terms of $AUROC$ for meta classification and in terms of R^2 for meta regression for CenterPoint on nuScenes. The tests for the meta classification and the meta regression are independent of each other, i.e., the selected features of the two saturation plots do not have to match. When using five selected features, the associated meta models perform already roughly as well as when using all features (LMD), i.e., 0.19 pp worse in terms of meta classification $AUROC$ and 0.92 pp worse in terms of meta regression R^2 . With ten features used, the respective differences with the results obtained by LMD are < 0.1 pp and thus negligible. The tests for the greedy selection heuristic for all network-dataset-combinations are shown in the appendix.

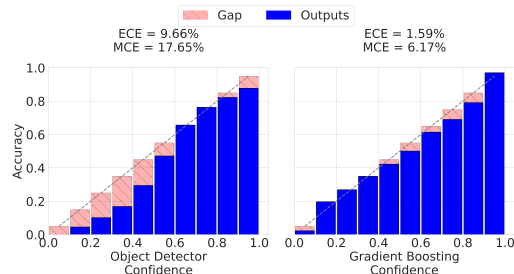


Fig. 8: Reliability plots of the score (left) and GB classifier for LMD (right) with calibration errors (ECE, MCE) for CenterPoint, nuScenes test dataset, score threshold 0.1 and IoU_{BEV} threshold 0.5.

Confidence Calibration. The score and the meta classifier confidences are divided into 10 confidence bins to evaluate their calibration errors. Figure 8 shows exemplary reliability plots for the object detector score and LMD based on a

GB classifier with corresponding expected (ECE [14]) and maximum calibration error (MCE [14]). The score is over-confident in the lower confidence ranges and well-calibrated in the upper confidence ranges, whereas the GB classifier for LMD is well-calibrated overall. This observation is also reflected in numerical calibration errors, as the GB classifier for LMD outperforms the score by 8.07 pp in terms of ECE and by 11.48 pp in terms of MCE. This indicates that LMD improves the statistical reliability of the confidence assignment.

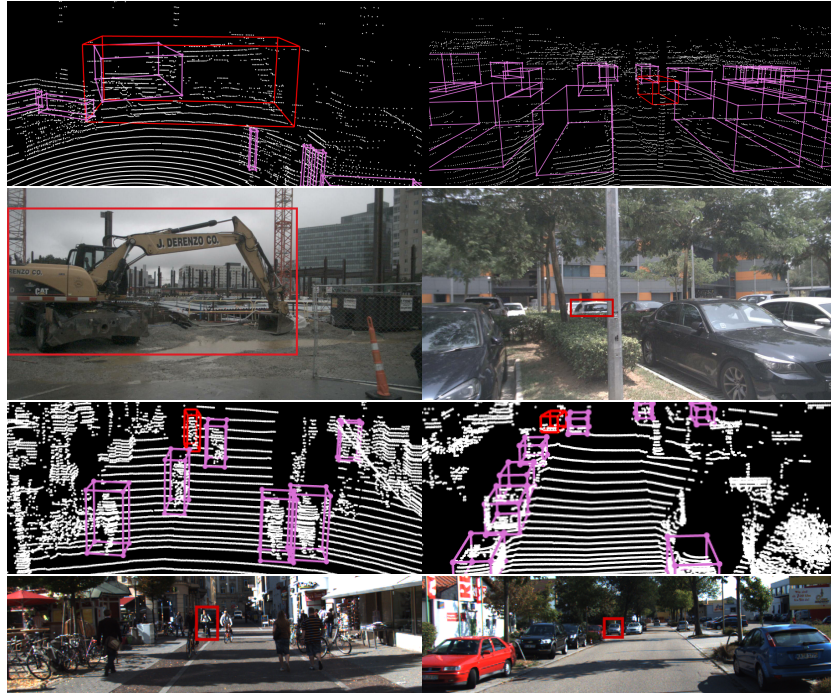


Fig. 9: Proposed annotation errors in nuScenes (top two) and KITTI (bottom two). Top images show point clouds with annotations in purple and the proposal in red. Camera images aid the evaluation.

Annotation Error Detection as an Application of Meta Classification. The task of annotations error detection with LMD is inspired by fig. 7. There are a number of predictions with $IoU_{BEV} = 0$ but with high predicted IoU_{BEV} . After looking at these FPs it has been noticed that the prediction itself is, in fact, correct and the corresponding ground truth is not. More precisely, incorrect ground truth corresponds to missing labels, labels with a wrong assigned class or the location of the annotation is inaccurate, i.e., the 3D bounding box is not correctly aligned with the point cloud. Annotation error detection with LMD works as follows: all FP predictions, i.e., predictions that have $IoU_{BEV} < 0.5$ with the ground truth,

are sorted by the predicted IoU_{BEV} in descending order across all images. Then, the first 100 predictions, i.e., the top 100 FPs with highest predicted IoU_{BEV} , are manually reviewed, see fig. 9 for examples of proposals by this method. In this case, a GB classifier is used to predict the box-wise IoU_{BEV} . We compare LMD against a score baseline which works in the same way, except that the FPs are sorted by the objectness score. As a random baseline, 100 randomly drawn FPs are considered for review which provides an insight into how well the respective test dataset is labeled. In general, if it was unclear whether an annotation error was present or not, this case was not marked as annotation error, i.e., the following numbers are a conservative (under-)estimation. LMD finds 43 annotation errors from 100 proposals and, in contrast, the score only 6 out of 100. Even the random baseline still finds 3 annotation errors, which indicates that there is a significant number of annotation errors in the nuScenes test dataset and that these can be found at far smaller effort with LMD than with the score. Annotation error detection counts for nuScenes and KITTI test datasets are shown in the appendix.

5 Conclusion

In this work we have introduced a purely post-processing-based uncertainty quantification method (LMD). A post-processing module, which is simple to fit and can be plugged onto any pre-trained Lidar object detector, allows for swift estimation of confidence (meta classification) and localization precision (meta regression) in terms of IoU_{BEV} at inference time. Our experiments show that separation of true and false predictions obtained from LMD is sharper than that of the base detector. Statistical reliability is significantly improved in terms of calibration of the obtained confidence scores and IoU_{BEV} is estimated to considerable precision at inference time, i.e., without knowledge of the ground truth. In addition to statistical improvement in decision making, we introduce a method for detecting annotation errors in real-world datasets based on our uncertainty estimation method. Error counts of hand-reviewed proposals which are shown for broadly used public benchmark datasets suggest a highly beneficial industrial use case of our method beyond improving prediction reliability. We also hope that our investigations will spark future research in the domains of light-weight uncertainty estimation and annotation error detection for large-scale datasets.

Acknowledgement

We gratefully acknowledge financial support by the state Ministry of Economy, Innovation and Energy of Northrhine Westphalia (MWIDE) and the European Fund for Regional Development via the FIS.NRW project BIT, grant no. EFRE-0400216, as well as “Projekt UnREAL”, grant no. 01IS22069, funded by the German Federal Ministry of Education and Research.

References

1. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuScenes: A Multimodal Dataset for Autonomous Driving. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11618–11628. IEEE, Seattle, WA, USA (Jun 2020). <https://doi.org/10.1109/CVPR42600.2020.01164>, <https://ieeexplore.ieee.org/document/9156412/>
2. Chen, H., Huang, Y., Tian, W., Gao, Z., Xiong, L.: MonoRUn: Monocular 3D Object Detection by Reconstruction and Uncertainty Propagation. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10374–10383. IEEE, Nashville, TN, USA (Jun 2021). <https://doi.org/10.1109/CVPR46437.2021.01024>, <https://ieeexplore.ieee.org/document/9578186/>
3. Davis, J., Goadrich, M.: The Relationship Between Precision-Recall and ROC Curves. Tech. rep., Department of Computer Sciences and Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison (2006), <https://minds.wisconsin.edu/bitstream/handle/1793/60482/TR1551.pdf>
4. Feng, D., Rosenbaum, L., Dietmayer, K.: Towards Safe Autonomous Driving: Capture Uncertainty in the Deep Neural Network For Lidar 3D Vehicle Detection. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). pp. 3266–3273 (Nov 2018). <https://doi.org/10.1109/ITSC.2018.8569814>, iSSN: 2153-0017
5. Feng, D., Rosenbaum, L., Timm, F., Dietmayer, K.: Leveraging Heteroscedastic Aleatoric Uncertainties for Robust Real-Time LiDAR 3D Object Detection. In: 2019 IEEE Intelligent Vehicles Symposium (IV). pp. 1280–1287 (Jun 2019). <https://doi.org/10.1109/IVS.2019.8814046>, iSSN: 2642-7214
6. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 3354–3361 (Jun 2012). <https://doi.org/10.1109/CVPR.2012.6248074>, iSSN: 1063-6919
7. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On Calibration of Modern Neural Networks
8. Havasi, M., Jenatton, R., Fort, S., Liu, J.Z., Snoek, J., Lakshminarayanan, B., Dai, A.M., Tran, D.: Training independent subnetworks for robust prediction (Aug 2021), <http://arxiv.org/abs/2010.06610>, arXiv:2010.06610 [cs, stat]
9. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: PointPillars: Fast Encoders for Object Detection From Point Clouds. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12689–12697. IEEE, Long Beach, CA, USA (Jun 2019). <https://doi.org/10.1109/CVPR.2019.01298>, <https://ieeexplore.ieee.org/document/8954311/>
10. Lin, W.H., Hauptmann, A.: Meta-classification: Combining Multimodal Classifiers. In: Zaïane, O.R., Simoff, S.J., Djeraba, C. (eds.) Mining Multimedia and Complex Data. pp. 217–231. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2003). <https://doi.org/10.1007/978-3-540-39666-6-14>
11. Meyer, G.P., Laddha, A., Kee, E., Vallespi-Gonzalez, C., Wellington, C.K.: LaserNet: An Efficient Probabilistic 3D Object Detector for Autonomous Driving. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12669–12678. IEEE, Long Beach, CA, USA (Jun 2019). <https://doi.org/10.1109/CVPR.2019.01296>, <https://ieeexplore.ieee.org/document/8953739/>

12. Meyer, G.P., Thakurdesai, N.: Learning an Uncertainty-Aware Object Detector for Autonomous Driving. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 10521–10527 (Oct 2020). <https://doi.org/10.1109/IROS45743.2020.9341623>, iSSN: 2153-0866
13. MMDetection3D Contributors: OpenMMLab’s Next-generation Platform for General 3D Object Detection (Jul 2020), <https://github.com/open-mmlab/mmdetection3d>, original-date: 2020-07-08T03:39:45Z
14. Naeini, M.P., Cooper, G.F., Hauskrecht, M.: Obtaining Well Calibrated Probabilities Using Bayesian Binning. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (2015)
15. Pitropov, M., Huang, C., Abdelzad, V., Czarnecki, K., Waslander, S.: LiDAR-MIMO: Efficient Uncertainty Estimation for LiDAR-based 3D Object Detection. In: 2022 IEEE Intelligent Vehicles Symposium (IV). pp. 813–820 (Jun 2022). <https://doi.org/10.1109/IV51971.2022.9827244>
16. Riedlinger, T., Rottmann, M., Schubert, M., Gottschalk, H.: Gradient-Based Quantification of Epistemic Uncertainty for Deep Object Detectors. In: 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 3910–3920. IEEE, Waikoloa, HI, USA (Jan 2023). <https://doi.org/10.1109/WACV56688.2023.00391>, <https://ieeexplore.ieee.org/document/10030773/>
17. Rottmann, M., Colling, P., Paul Hack, T., Chan, R., Hüger, F., Schlicht, P., Gottschalk, H.: Prediction Error Meta Classification in Semantic Segmentation: Detection via Aggregated Dispersion Measures of Softmax Probabilities. In: 2020 International Joint Conference on Neural Networks (IJCNN). pp. 1–9 (Jul 2020). <https://doi.org/10.1109/IJCNN48605.2020.9206659>, iSSN: 2161-4407
18. Rottmann, M., Maag, K., Chan, R., Hüger, F., Schlicht, P., Gottschalk, H.: Detection of False Positive and False Negative Samples in Semantic Segmentation. In: 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE). pp. 1351–1356 (Mar 2020). <https://doi.org/10.23919/DATE48585.2020.9116288>, iSSN: 1558-1101
19. Schubert, M., Kahl, K., Rottmann, M.: MetaDetect: Uncertainty Quantification and Prediction Quality Estimates for Object Detection. In: 2021 International Joint Conference on Neural Networks (IJCNN). pp. 1–10 (Jul 2021). <https://doi.org/10.1109/IJCNN52387.2021.9534289>, iSSN: 2161-4407
20. Stanley, T.D., Jarrell, S.B.: Meta-Regression Analysis: A Quantitative Method of Literature Surveys. *Journal of Economic Surveys* **19**(3), 299–308 (2005). <https://doi.org/10.1111/j.0950-0804.2005.00249.x>, <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.0950-0804.2005.00249.x>, [_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.0950-0804.2005.00249.x](https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.0950-0804.2005.00249.x)
21. Yan, Y., Mao, Y., Li, B.: Second: Sparsely embedded convolutional detection. *Sensors* **18**(10), 3337 (2018), publisher: MDPI
22. Yang, B., Luo, W., Urtasun, R.: PIXOR: Real-time 3D Object Detection from Point Clouds. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7652–7660. IEEE, Salt Lake City, UT, USA (Jun 2018). <https://doi.org/10.1109/CVPR.2018.00798>, <https://ieeexplore.ieee.org/document/8578896/>
23. Yang, Q., Chen, H., Chen, Z., Su, J.: Uncertainty Estimation for Monocular 3D Object Detectors in Autonomous Driving. In: 2021 6th International Conference on Robotics and Automation Engineering (ICRAE). pp. 55–59 (Nov 2021). <https://doi.org/10.1109/ICRAE53653.2021.9657820>

24. Yin, T., Zhou, X., Krahenbuhl, P.: Center-based 3D Object Detection and Tracking. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11779–11788. IEEE, Nashville, TN, USA (Jun 2021). <https://doi.org/10.1109/CVPR46437.2021.01161>, <https://ieeexplore.ieee.org/document/9578166/>