Uncertainty-Aware Scene Understanding via Efficient Sampling-Free Confidence Estimation

Hanieh Shojaei Miandashti
Institute of Cartography and Geoinformatics
Leibniz University Hannover, Germany
Hanieh.Shojaei@ikg.uni-hannover.de

Qianqian Zou
Institute of Cartography and Geoinformatics
Leibniz University Hannover, Germany
qianqian.zou@ikg.uni-hannover.de

Claus Brenner
Institute of Cartography and Geoinformatics
Leibniz University Hannover, Germany
claus.brenner@ikg.uni-hannover.de

Abstract:
Reliable scene understanding requires not only accurate predictions but also well-calibrated confidence estimates to ensure reliable uncertainty estimation, especially in safety-critical domains like autonomous driving. In this context, semantic segmentation of LiDAR points supports real-time 3D scene understanding, where reliable uncertainty estimates help identify potentially erroneous predictions. While most existing calibration approaches focus on modeling epistemic uncertainty, they often overlook aleatoric uncertainty arising from measurement inaccuracies, which is especially prevalent in LiDAR data and essential for real-world deployment. In this work, we introduce a sampling-free approach for estimating well-calibrated confidence values by explicitly modeling aleatoric uncertainty in semantic segmentation, achieving alignment with true classification accuracy and reducing inference time compared to sampling-based methods. Evaluated on the real-world SemanticKITTI benchmark, our approach achieves 1.70% and 1.33% Adaptive Calibration Error (ACE) in semantic segmentation of LiDAR data using RangeViT and SalsaNext models, and is more than one order of magnitude faster than the comparable baseline. Furthermore, reliability diagrams reveal that our method produces underconfident rather than overconfident predictions — an advantageous property in safety-critical systems.

Keywords: Confidence Calibration in Deep Learning, Aleatoric Uncertainty Estimation, Reliable Semantic Segmentation of LiDAR Point Clouds

1 Introduction

In safety-critical domains such as autonomous driving, deep neural networks (DNNs) must ensure both high accuracy and well-calibrated confidence estimates to support reliable uncertainty estimation, allowing the system to recognize when it is likely to be wrong and act conservatively. For 3D scene understanding from LiDAR point clouds, this means identifying points where the predicted semantic labels may be unreliable. Ideally, confidence values should align with the true likelihood of correctness; however, modern DNNs often produce overconfident outputs, failing to capture inherent uncertainties in data and model predictions [1, 2, 3]. Such miscalibration is especially concerning in safety-critical settings, where high-confidence errors can lead to unsafe decisions. Proper calibration is thus essential for building trustworthy autonomous systems.

A variety of approaches have been proposed to improve confidence calibration, including post-hoc techniques and uncertainty quantification methods [4, 5, 6]. Among post-hoc methods, temperature

9th Conference on Robot Learning (CoRL 2025), Seoul, Korea.

scaling is widely used due to its simplicity and effectiveness in calibrating DNNs [1]. However, uncertainty-aware approaches such as deep ensembles [7] and Monte Carlo Dropout (MC dropout) [8] have shown better calibration performance [2]. Recent studies have further shown that temperature scaling becomes ineffective when class distributions overlap significantly, particularly as the number of classes increases [9]. To address this limitation, we propose a method that explicitly models aleatoric uncertainty by representing each class logit as a Gaussian distribution and incorporating distributional overlap into confidence estimation. Confidence is estimated as the probability that the predicted class yields a higher sampled logit than all competing classes, effectively quantifying the likelihood of a correct prediction. While this probability is typically approximated via inefficient Monte Carlo sampling, we introduce a closed-form lower bound that eliminates the need for sampling in multi-class settings, making the approach well-suited for real-time applications.

Reliable confidence calibration must account for both aleatoric uncertainty and epistemic uncertainty. Aleatoric uncertainty arises from irreducible inherent noise in the data, such as LiDAR sensor inaccuracies, varying point cloud density with distance, environmental variability, and surface reflectivity. In contrast, epistemic uncertainty stems from limited model knowledge and can be reduced with more data [10]. While most existing methods focus primarily on epistemic uncertainty [7, 8, 11, 12], our approach explicitly incorporates aleatoric uncertainty directly into the confidence estimation process, leading to better calibration on complex, noisy input data like LiDAR point clouds. That is, we integrate epistemic uncertainty with our aleatoric uncertainty to produce well-calibrated confidence estimates.

Our main contribution is a novel confidence estimation method that accounts for the overlap between logit distributions to compute well-calibrated confidence values. We make three key claims. **First**, our proposed approach generates confidence values that closely approximate true confidence scores, effectively calibrated against the actual classification accuracy. Additionally, our method produces underconfident rather than overconfident values, making it particularly valuable for safety-critical decision-making. **Second**, our approach outperforms the comparable approaches (e.g., temperature scaling) by accounting for the underlying data distribution. Moreover, when combined with epistemic uncertainty, it achieves the highest calibration performance. **Third**, our sampling-free approach reduces inference time compared to sampling methods such as logit-sampling [13] while maintaining confidence calibration and classification accuracy. These proposed contributions are validated through experiments conducted on benchmark datasets of SemanticKITTI [14] and nuScenes [15], evidenced by our performance on the Adaptive Calibration Error (ACE) [16] metric and further observed in the reliability diagram [1].

2 Related Works

2.1 Confidence calibration

Confidence values were first introduced as Maximum Class Probability (MCP), the highest probability in the softmax distribution, based on the assumption that correctly classified samples generally have higher MCP than misclassified and out-of-distribution examples [17]. However, subsequent studies have identified major limitations in MCP: it often produces overly confident estimates, highlighting the need for confidence calibration [18, 19, 1, 20].

Previous research on confidence calibration of deep learning models typically falls into two categories. The first involves post-hoc methods that adjust classifier outputs by rescaling the logits without retraining the model [1, 6, 21]. Temperature scaling [1] exemplifies this by recalibrating model logits using a single parameter optimized on a validation set.

While temperature scaling and its variants are widely used for post-hoc calibration, they primarily adjust predicted probabilities without addressing the underlying sources of miscalibration, such as model uncertainty or data noise. These methods do not modify the model itself and therefore fail to correct overconfidence that stems from overlapping or noisy class distributions. Recent work by Chidambaram and Ge [9] has shown, both theoretically and empirically, that temperature scaling be-

comes increasingly ineffective as class overlap grows, and can asymptotically perform no better than random guessing in multi-class settings. In contrast, the second category incorporates uncertainty directly into the training phase, enhancing models' inherent ability to account for data variability.

2.2 Uncertainty estimation

Methods such as Bayesian neural networks (BNNs) [22] and evidential deep learning (EDL) [23] equip models to inherently represent uncertainty, providing a more fundamental solution to confidence calibration challenges. While EDL calculates total uncertainty without differentiating between epistemic and aleatoric, BNNs specifically model epistemic uncertainty by placing a prior distribution over parameters of a model and approximating the posterior distribution through Bayesian inference [22, 24]. However, due to the often intractable nature of exact inference, variational methods such as Bayes by backprop [25] employ an evidence-based lower bound for approximating the posterior distribution. Several tractable methods for estimating epistemic uncertainty have emerged in recent years [26, 27, 12], including MC dropout, which applies dropout during inference, and deep ensembles, which approximate the posterior distribution by training multiple networks with different initializations. However, although these models effectively estimate confidence by averaging softmax outputs from multiple instances, they may still require additional calibration to better align softmax probabilities. This limitation arises because they primarily rely on the model's softmax values without explicitly accounting for aleatoric uncertainty.

To accurately calibrate softmax outputs by accounting for the underlying true distribution, the logit-sampling approach proposed a method that assumes a Gaussian distribution for the logits of each class [13]. However, during its inference, the need to perform Monte Carlo sampling across each distribution to compute calibrated confidence introduces additional computational overhead and increases inference time. Our proposed approach lies in this line of work and quantifies aleatoric uncertainty from Gaussian logit distributions without sampling, making it suitable for real-time applications with many classes.

2.3 Semantic segmentation of LiDAR point clouds

3D scene perception using deep learning-based semantic segmentation of LiDAR point clouds can be classified into two categories based on their underlying 3D representations [28]: The first category includes point-wise methods that process 3D data directly, including raw 3D point-based architectures [29, 30, 31, 32] and voxel-based networks [33, 34, 35, 36, 37]. While voxel-based methods convert unordered point clouds into structured 3D grids—enabling the use of 3D convolutions to capture geometric features—they are often computationally intensive. In contrast, the second category comprises projection-based methods, which convert 3D point clouds into 2D representations, either as bird's-eye view maps [38, 39] or spherical range-view images (panoramic view) [40, 41, 42, 43, 44]. These methods benefit from the structured 2D image representations, enabling the use of 2D convolutional neural networks (CNNs) and vision transformers (ViTs), which are more computationally efficient.

In this work, we adopt SalsaNext [40] and RangeViT [44] as projection-based architectures for LiDAR semantic segmentation, both of which have demonstrated state-of-the-art performance on standard benchmark datasets.

3 Methodology

Our proposed method is based on the Gaussian distributions over the logits by directly estimating a mean (μ_i) and variance (σ_i^2) for each of the C classes before applying the softmax function. During training, we sample T logit vectors using the reparameterization trick, defined as $\mathbf{z}^{(t)} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}^{(t)}$, where $\boldsymbol{\epsilon}^{(t)} \sim \mathcal{N}(0, \mathbf{I})$ is a standard Gaussian noise vector and $\odot$ denotes element-wise multiplication. For each sampled logit vector, we compute softmax probabilities as $\mathbf{p}^{(t)} =$

softmax($\mathbf{z}^{(t)}$), and use their average, $\bar{\mathbf{p}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{p}^{(t)}$, as the final prediction input to the loss function.

During inference, unlike sampling-based approaches, our method performs a single feed-forward pass and selects the class with the highest predicted mean. We directly compute its well-calibrated confidence value by evaluating the probability that logit scores from this class exceed those of competing classes. This sampling-free formulation is detailed in the following.

3.1 Confidence computation

3.1.1 Exact and approximate computation

As usual, one would predict the class whose predicted mean is maximal. Let there be C classes, for each of which a Gaussian distribution $\mathcal{N}(\cdot|\mu_i, \sigma_i^2)$, $1 \leq i \leq C$ is predicted. Then, without loss of generality, we may assume $\mu_1 \geq \mu_i$ for $i \geq 2$, so that class 1 is selected as the predicted class. The confidence is then defined as the probability that this selection is correct, which, given random variables $X_i \sim \mathcal{N}(\cdot|\mu_i, \sigma_i^2)$, is $P(X_1 \geq \{X_i\}_{i\geq 2})$, equivalent to $P(X_1 \geq \max_{i\geq 2} X_i)$.

As there is no closed-form solution to compute this probability in the general case, it may be approximated by computing a relative frequency by simulation. Similar to the training phase, C samples X_i, $1 \leq i \leq C$ are drawn (one from each Gaussian), and it is determined if X_1 is largest. From repeating this experiment N times, the relative frequency of cases $X_1 \geq \max_{i\geq 2} X_i$ is computed. This method requires NC draws.

It is easily seen that the required probability is given by

$$P(X_1 \geq \max_{i\geq 2} X_i) = \int_{-\infty}^{+\infty} \varphi_1(x) \prod_{i=2}^{C} \Phi_i(x) dx, \tag{1}$$

where we have used $\varphi_i(x) = \mathcal{N}(x|\mu_i, \sigma_i^2)$ for the Gaussian densities and $\Phi_i(x)$ for their associated cumulative distribution functions. This integral can be approximated via Monte Carlo simulation using

$$P(X_1 \geq \max_{i\geq 2} X_i) \approx \frac{1}{N}\sum_{k=1}^{N}\prod_{i=2}^{C} \Phi_i(x_k), \tag{2}$$

with $x_k \sim \varphi_1$, requiring only N samples to be drawn (which in fact can be re-used by scaling and shifting a fixed set of samples).

3.1.2 Sampling-free confidence quantification

For the special case of two classes, a closed-form solution can be given, because $P(X_1 \geq X_2) = P(Z \geq 0)$ with $Z := X_1 - X_2$. As $Z \sim \mathcal{N}(\cdot|\mu_1 - \mu_2, \sigma_1^2 + \sigma_2^2)$ it follows that

$$P(X_1 \geq X_2) = \Phi(\mu_1 - \mu_2|0, \sigma_1^2 + \sigma_2^2) =: \Phi_{1,2}, \tag{3}$$

which unfortunately does not extend readily to $C > 2$ classes. However, using pairwise $\Phi_{1,i}$, $i \geq 2$, the following lower bound holds

$$P(X_1 \geq \max_{i\geq 2} X_i) \geq \prod_{i=2}^{C} \Phi_{1,i}, \tag{4}$$

which follows from the fact that the integral in Equation 1 is the expected value $\mathbb{E}[\prod_{i=2}^{C} \Phi_i(X)]$ under the distribution $X \sim \varphi_1$, and since all $\Phi_i(x)$ are strictly monotonically increasing functions of x, their covariance is non-negative, from which it follows that $\mathbb{E}[\Phi_i(X)\Phi_j(X)] \geq \mathbb{E}[\Phi_i(X)]\mathbb{E}[\Phi_j(X)]$, yielding Equation 4. For products of more than two terms, this follows by recursive application, noting that any product of Gaussian cumulative distribution functions is strictly monotonically increasing as well. A detailed derivation of Equation 4 is provided in Section A.1.

To conclude, given the predicted class distributions, a lower bound for the confidence can be computed by simply evaluating $C - 1$ 'pairwise' cumulative distribution functions, requiring no sampling. To give some intuition, if class 1 is a clear winner, the confidence will be 1. If the winner class is challenged only by one alternative class, the lower bound Equation 4 reduces to Equation 3, and the bound is exact. If more than one other class challenges the winner class, the lower bound will underestimate the confidence.

In our experiments, the proposed lower bound exhibits only a negligible difference when compared to the exact formulation. As demonstrated in Section 5.1, this lower bound provides a close approximation to the true value obtained through Monte Carlo integration, while exhibiting a slightly underconfident behavior.

4 Experimental setup

4.1 Datasets and network architectures

We evaluate our approach on two widely used LiDAR semantic segmentation benchmarks—SemanticKITTI [14] and nuScenes [15]. Each 3D scan from SemanticKITTI and nuScenes was converted based on their LiDAR beams into a $[64 \times 2048 \times 5]$ and $[32 \times 2048 \times 5]$ spherical range-view image, respectively. Each image contains five channels corresponding to 3D point coordinates (x, y, z), intensity, and range values, serving as input for semantic segmentation.

We employ SalsaNext [40] as a CNN model, which adopts a U-Net encoder-decoder architecture enhanced with ResNet blocks for efficient feature extraction, and RangeViT, a transformer-based model that exploits Vision Transformers (ViTs) for LiDAR semantic segmentation.

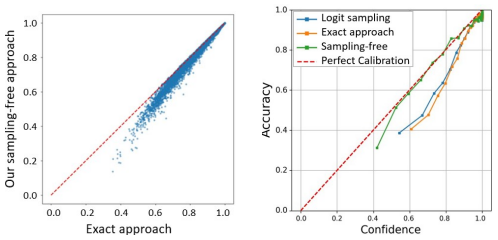
For the loss function, we adopt an equally weighted combination of the multi-class focal loss [45], with a focusing parameter $\gamma = 2.0$ and the Lovász-Softmax loss [46], in order to jointly improve confidence calibration and segmentation accuracy. All models are trained for 60 epochs using the Adam optimizer with an initial learning rate of 0.01, which is decayed by a factor of 0.01 after each epoch.

4.2 Comparative methods

To evaluate our proposed method, we use MCP as the baseline uncalibrated confidence measure, and apply temperature scaling, logit-sampling, and our approach for confidence calibration. Each method is further combined with epistemic uncertainty modeling using deep ensembles (DE) and MC dropout. Additionally, we assess EDL as a competitive calibration strategy. The evaluation of all methods is summarized in Table 1 of Section 5.1.

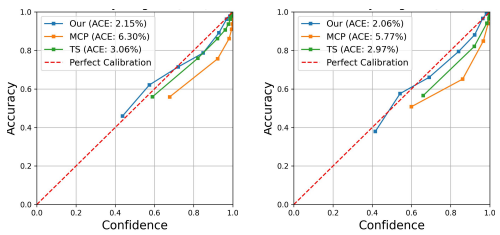
4.3 Evaluation metrics

Classification performance is evaluated using the mean Intersection over Union (mIoU) metric. To evaluate the calibration of the confidence values for predicted classes, we use the ACE [16], complemented by reliability diagrams [1], which illustrate whether the model is underconfident or overconfident—information that ACE does not provide, as it only quantifies the absolute deviation from perfect calibration. In contrast to the Expected Calibration Error (ECE) [47], ACE assigns equal weight to each bin in the reliability diagram, defined as $\text{ACE} = \frac{1}{M} \sum_{m=1}^{M} |\text{Conf}_m - \text{Acc}_m|$, where M represents the number of non-empty bins, Conf_m is the average confidence within bin m, and Acc_m is the corresponding average accuracy.



(a) Ours vs. exact (b) Reliability diagram

Figure 1: Comparison of confidence estimation methods: (a) Scatter plot shows minimal difference between exact and sampling-free confidences; (b) Reliability diagram indicates our method produces better calibration compared to baselines.



(a) SalsaNext (b) RangeViT

Figure 2: Reliability diagrams comparing calibration of our method against temperature scaling (TS) and uncalibrated (MCP) models on SemanticKITTI validation set using SalsaNext and RangeViT. Our method shows better calibration (closer to perfect calibration).

5 Experiments

5.1 Comparative analysis of sampling-based and sampling-free confidence computations

To support our first claim that the sampling-free lower bound approach closely estimates the true confidence values, we compare these confidence values with those obtained from the Monte Carlo integration. They are computed based on Equation 4 and Equation 2, respectively. The scatter plot in Figure 1a illustrates this comparison on a subset of validation samples of SemanticKITTI with the SalsaNext model, with x-axis representing the exact confidence values and the y-axis showing the lower bound estimation. The red dashed line denotes the ideal $y = x$ line, where the lower bound would match the exact computation.

Figure 1a shows that the lower bound estimation predominantly aligns closely with or slightly underestimates the exact values, evidenced by the clustering of points below the $y = x$ line. This pattern indicates that the lower bound estimation tends to behave conservatively, often yielding slightly underestimated confidence values across a broad range.

Additionally, Figure 1b contrasts the true classification accuracy (y-axis) with the exact confidence computation, the sampling-free lower bound approach and the logit-sampling baseline. Here, the sampling-free lower bound approach, depicted by the green line, aligns more closely with the ideal red line and consistently shows more conservative confidence estimates relative to the logit-sampling baseline and the exact values. This visualization highlights the conservative nature of our sampling-free approach, supporting our first claim that it is not only well-calibrated but also reliably underconfident for improved safety.

5.2 Confidence calibration analysis

In this section, we provide experimental evidence supporting our second claim that our approach produces better-calibrated confidence estimates than temperature scaling. Figure 2 presents reliability diagrams of our confidence calibration method across two backbone networks—SalsaNext (2a) and RangeViT (2b)—on the SemanticKITTI dataset, alongside the uncalibrated model (MCP) and the temperature-scaled variant (TS). While temperature scaling improves the calibration of MCP, it still deviates noticeably from the perfect calibration (red dashed line). In contrast, our method consistently remains closer to perfect calibration and displays mild underconfidence in the 0.4 to 0.7 confidence range. On SalsaNext, our approach achieves an ACE of 2.15%, outperforming TS (3.06%) and MCP (6.30%). Similarly, for RangeViT, our method attains an ACE of 2.06%, compared to 2.97% for TS and 5.77% for MCP. These results confirm the superior calibration performance of our approach across both architectures.

To further support our second claim, that combining epistemic uncertainty with aleatoric uncertainty achieves the highest calibration performance, we conduct further analysis by combining our aleatoric confidence estimation with two epistemic modeling approaches: DE [7] and MC dropout [8]. Results in Table 1 demonstrate that once the epistemic uncertainty is incorporated, both our approach and the logit-sampling (LS) method, paired with either DE or MC dropout, consistently outperformed all other methods in terms of ACE, achieving the lowest calibration error across both semantic segmentation networks (1.70% and 1.21% on RangeViT and SalsaNext for SemanticKITTI and 1.78% on RangeViT for nuScenes). These results highlight the significance of jointly modeling both aleatoric and epistemic uncertainty for effective confidence calibration.

Overall, Table 1 shows that the lowest calibration errors are achieved by methods that incorporate aleatoric uncertainty, with further improvements when epistemic uncertainty is jointly modeled—whether via DE or MC dropout. These approaches consistently outperform EDL, temperature scaling and uncalibrated baselines. Notably, for both datasets with RangeViT, the best ACE (1.70% and 1.73%) is achieved by our sampling-free method combined with DE, a trend that also holds for SalsaNext, where the same combination yields an ACE of 1.33%. Qualitative results detailed in Section A.2 demonstrate that our proposed approach, combined with DE, estimates predictive uncertainty that closely follows the error map, thereby producing uncertainty-aware semantic segmentation of LiDAR scans.

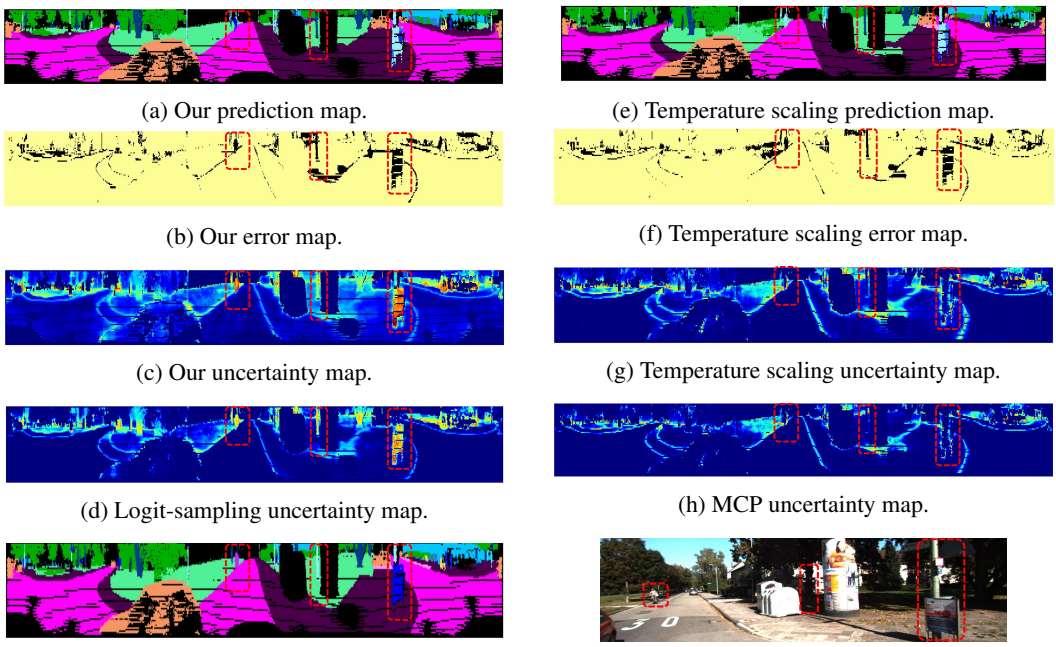
Qualitative results in Figure 3 support our findings. Three misclassified objects, indicated by red dashed boxes, exhibit low uncertainty in the maps produced by the uncalibrated model (MCP; Figure 3h) and temperature scaling (Figure 3g). In contrast, our proposed method (Figure 3c) and the competitive logit-sampling approach (Figure 3d) assign higher uncertainty to these samples and demonstrate improved accuracy, as evident from the comparison of the corresponding error maps. Additional qualitative examples are provided in Section A.2.

Table 1: Comparative analysis of inference time (s)↓, mIoU (%)↑, and ACE (%)↓ across confidence calibration methods on the SemanticKITTI and nuScenes validation sets using SalsaNext and RangeViT. The best-performing results are highlighted in bold, and the second-best are shown in blue.

Method	Uncertainty Type		RangeViT (SemanticKITTI)			SalsaNext (SemanticKITTI)			RangeViT (nuScenes)		
	Aleatoric	Epistemic	mIoU	ACE	Time	mIoU	ACE	Time	mIoU	ACE	Time
MCP			58.40	5.77	0.09	50.06	6.30	0.12	73.81	3.71	0.04
MCP + DE		✓	60.24	4.63	0.48	**51.80**	5.01	0.67	74.21	2.90	0.26
MCP + MC dropout		✓	59.93	4.71	0.54	51.23	4.60	0.63	73.88	3.27	0.31
TS			58.40	2.97	0.11	50.06	3.06	0.15	73.81	2.66	0.06
TS + DE		✓	60.24	2.21	0.61	**51.80**	2.84	0.81	74.21	2.31	0.34
TS + MC dropout		✓	59.93	2.97	0.67	51.23	2.26	0.73	73.88	2.23	0.40
LS	✓		60.21	2.11	3.61	51.03	2.03	4.80	74.92	2.11	2.01
LS + DE	✓	✓	**60.54**	1.83	20.01	51.42	**1.21**	26.00	**75.01**	1.78	12.01
LS + MC dropout	✓	✓	60.33	1.81	27.11	51.70	1.63	29.60	74.95	2.01	14.00
Our approach	✓		60.21	2.06	0.25	51.03	2.15	0.28	74.92	2.18	0.11
Our approach + DE	✓	✓	**60.54**	**1.70**	1.33	51.42	1.33	1.46	**75.01**	**1.73**	0.73
Our approach + MC dropout	✓	✓	60.33	1.95	1.48	51.70	1.91	1.90	74.95	1.94	0.88
EDL	✓	✓	57.33	5.83	0.18	48.30	5.32	0.15	68.01	2.91	0.09

5.3 Inference time analysis

To validate our third claim, that our proposed confidence estimation method reduces inference time compared to sampling-based approaches for modeling aleatoric uncertainty, Table 2 compares inference time and floating point operations (FLOPs) between our sampling-free method and the logit-sampling approach. As shown, our method achieves a substantial reduction in inference time, decreasing it by a factor of 15 times for RangeViT and 18 times for SalsaNext, while incurring only minimal computational overhead. Specifically, the logit-sampling method adds 6.82G FLOPs to both models due to 50 times sampling per pixel, whereas our method increases the total FLOPs by just 0.07G FLOPs through pairwise CDF computations. All results are measured on a GeForce RTX 3060 GPU.



(a) Our prediction map.

(e) Temperature scaling prediction map.

(b) Our error map.

(f) Temperature scaling error map.

(c) Our uncertainty map.

(g) Temperature scaling uncertainty map.

(d) Logit-sampling uncertainty map.

(h) MCP uncertainty map.

Ground truth.

Camera image.

Figure 3: Qualitative comparison of uncertainty maps from the logit-sampling baseline and our sampling-free method (a-d). Both align with misclassifications, but our method shows higher uncertainty in error-prone regions, reflecting more underconfident estimation. Uncertainty maps are compared to those from temperature scaling and an uncalibrated model (e-h), which has not detected those misclassifications as high-uncertainty regions. outlier, car, road, sidewalk, building, fence, vegetation, trunk, terrain, pole, traffic-sign.

Table 2: Comparison of inference time and FLOPs between our proposed sampling-free approach and logit-sampling for aleatoric uncertainty consideration in confidence estimation on the validation set of SemanticKITTI.

Method	SalsaNext		RangeViT	
	Inference time	**FLOPs**	**Inference time**	**FLOPs**
Original model	0.12 ms	62.62G	0.09 ms	52.01G
+ logit-sampling approach	4.80 ms	69.44G	3.61 ms	58.83G
+ our sampling-free approach	0.28 ms	62.69G	0.25 ms	52.08G

6 Conclusion

We have developed a method to estimate the likelihood that the predicted class is correct, by examining the distributions of all possible class outcomes. We validated our sampling-free confidence estimation method on public datasets for LiDAR scene semantic segmentation, a field where safety-critical responses and real-time processing for large-scale data are crucial. Our comprehensive analysis comparing the lower bound confidences with the exact ones approximated through Monte Carlo integration, demonstrates a negligible discrepancy, confirming the robustness of our sampling-free lower bound confidence calibration approach for practical applications. Furthermore, when compared to the baseline approaches during inference, our proposed method consistently generates well-calibrated confidence values, exhibiting low ACE, calibrated reliability diagrams and fast inference. Moreover, our proposed method often tends to be slightly underconfident across a broader range of regions. In conclusion, our proposed approach effectively performs semantic segmentation, ensuring well-calibrated confidence computation and efficient performance, while also providing detailed uncertainty maps for pixel-wise semantic segmentation of LiDAR data.

Acknowledgments

This project is supported by the German Research Foundation (DFG), as part of the Research Training Group i.c.sens, GRK 2159, 'Integrity and Collaboration in Dynamic Sensor Networks'.

7 Limitations

Our results confirm that while our approach performs accurately and confidently on major classes well-represented during training—such as cars, roads, sidewalks, buildings, fences, and vegetation—it may struggle to distinguish between classes with similar features, occasionally confusing poles with thin trunks or bicycles with bicyclists. The proximity between Gaussian distributions does not necessarily result in misclassification if the predicted class maintains the highest mean. However, the closeness to another class's distribution increases predictive uncertainty, leading the model to be underconfident even when the object is correctly classified.

Another limitation of this approach is the increase in training time compared to the original model, as it predicts the mean and variance to model Gaussian distributions over the logits for each class, which may require additional computational cost.

As a direction for future work, it would be valuable to investigate the use of normalizing flows as a more flexible alternative to directly predicting mean and variance for modeling class-conditional distributions. Additionally, evaluating the proposed approach for out-of-distribution detection and assessing its robustness under domain shift scenarios would further demonstrate its applicability to real-world settings.

References

[1] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

[2] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32, 2019.

[3] D.-B. Wang, L. Feng, and M.-L. Zhang. Rethinking calibration of deep neural networks: Do not be afraid of overconfidence. *Advances in Neural Information Processing Systems*, 34: 11809–11820, 2021.

[4] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, 56(Suppl 1):1513–1589, 2023.

[5] C. Wang. Calibration in deep learning: A survey of the state-of-the-art. *arXiv preprint arXiv:2308.01222*, 2023.

[6] 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*, 32, 2019.

[7] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[8] Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[9] M. Chidambaram and R. Ge. On the limitations of temperature scaling for distributions with overlaps. *arXiv preprint arXiv:2306.00740*, 2023.

[10] Y. Gal et al. Uncertainty in deep learning. *thesis, University of Cambridge*, 2016.

[11] J. Mukhoti, A. Kirsch, J. van Amersfoort, P. H. Torr, and Y. Gal. Deep deterministic uncertainty: A new simple baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24384–24394, 2023.

[12] I. Osband, Z. Wen, S. M. Asghari, V. Dwaracherla, M. Ibrahimi, X. Lu, and B. Van Roy. Epistemic neural networks. *Advances in Neural Information Processing Systems*, 36:2795–2823, 2023.

[13] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

[14] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9297–9307, 2019.

[15] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[16] J. Nixon, M. W. Dusenberry, L. Zhang, G. Jerfel, and D. Tran. Measuring calibration in deep learning. In *CVPR workshops*, volume 2, 2019.

[17] D. Hendrycks and K. Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

[18] H. Jiang, B. Kim, M. Guan, and M. Gupta. To trust or not to trust a classifier. *Advances in neural information processing systems*, 31, 2018.

[19] C. Corbière, N. Thome, A. Bar-Hen, M. Cord, and P. Pérez. Addressing failure prediction by learning model confidence. *Advances in Neural Information Processing Systems*, 32, 2019.

[20] S. Thulasidasan, G. Chennupati, J. A. Bilmes, T. Bhattacharya, and S. Michalak. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. *Advances in neural information processing systems*, 32, 2019.

[21] B. Zadrozny and C. Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 694–699, 2002.

[22] R. M. Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

[23] M. Sensoy, L. Kaplan, and M. Kandemir. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems*, 31, 2018.

[24] C. Louizos and M. Welling. Multiplicative normalizing flows for variational bayesian neural networks. In *International Conference on Machine Learning*, pages 2218–2227. PMLR, 2017.

[25] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pages 1613–1622. PMLR, 2015.

[26] A. Kendall, V. Badrinarayanan, and R. Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *arXiv preprint arXiv:1511.02680*, 2015.

[27] K. Lee, H. Lee, K. Lee, and J. Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *arXiv preprint arXiv:1711.09325*, 2017.

[28] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(12): 4338–4364, 2020.

[29] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[30] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, and A. Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11108–11117, 2020.

[31] G. Puy, A. Boulch, and R. Marlet. Using a waffle iron for automotive point cloud semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3379–3389, 2023.

[32] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019.

[33] C. Choy, J. Gwak, and S. Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019.

[34] R. Cheng, R. Razani, E. Taghavi, E. Li, and B. Liu. 2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12547–12556, 2021.

[35] H. Tang, Z. Liu, S. Zhao, Y. Lin, J. Lin, H. Wang, and S. Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *European conference on computer vision*, pages 685–702. Springer, 2020.

[36] X. Zhu, H. Zhou, T. Wang, F. Hong, Y. Ma, W. Li, H. Li, and D. Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9939–9948, 2021.

[37] J. Huang and S. You. Point cloud labeling using 3d convolutional neural network. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2670–2675. IEEE, 2016.

[38] Q. Chen, S. Vora, and O. Beijbom. Polarstream: Streaming object detection and segmentation with polar pillars. *Advances in Neural Information Processing Systems*, 34:26871–26883, 2021.

[39] Y. Zhang, Z. Zhou, P. David, X. Yue, Z. Xi, B. Gong, and H. Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9601–9610, 2020.

[40] T. Cortinhal, G. Tzelepis, and E. Erdal Aksoy. Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds. In *Advances in Visual Computing: 15th International Symposium, ISVC 2020, San Diego, CA, USA, October 5–7, 2020, Proceedings, Part II 15*, pages 207–222. Springer, 2020.

[41] C. Xu, B. Wu, Z. Wang, W. Zhan, P. Vajda, K. Keutzer, and M. Tomizuka. Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 1–19. Springer, 2020.

[42] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 4213–4220. IEEE, 2019.

[43] L. Kong, Y. Liu, R. Chen, Y. Ma, X. Zhu, Y. Li, Y. Hou, Y. Qiao, and Z. Liu. Rethinking range view representation for lidar segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 228–240, 2023.

[44] A. Ando, S. Gidaris, A. Bursuc, G. Puy, A. Boulch, and R. Marlet. Rangevit: Towards vision transformers for 3d semantic segmentation in autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5240–5250, 2023.

[45] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[46] M. Berman, A. R. Triki, and M. B. Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4413–4421, 2018.

[47] M. P. Naeini, G. Cooper, and M. Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29, 2015.

[48] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Toronto, ON, Canada*, 2009.

[49] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

A Supplementary Material

A.1 Derivation of the lower bound formula

This derives the lower bound formula (Equation 4) of the main paper.

Given two Gaussians, $X \sim \mathcal{N}(x|\mu_a, \sigma_a^2) =: \varphi_a(x)$ and $Y \sim \mathcal{N}(\mu_b, \sigma_b^2)$, the probability $P(X > Y)$ is easily seen to be $P(X > Y) = \mathbb{E}_{\varphi_a}[\Phi_b(X)] = \Phi(\mu_a - \mu_b|0, \sigma_a^2 + \sigma_b^2)$, where $\varphi(x)$ and $\Phi(x)$ denote the Gaussian PDF and CDF, respectively, and $\mathbb{E}_{\varphi_a}[\cdot]$ is the expectation over the distribution φ_a.

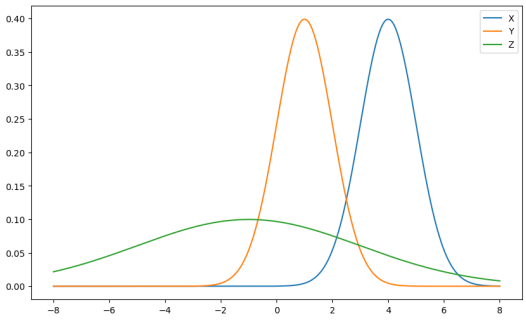


Figure 4: Illustration for three classes $X \sim \mathcal{N}(\mu_a = 4, \sigma_a^2 = 1)$, $Y \sim \mathcal{N}(\mu_b = 1, \sigma_b^2 = 1)$, and $Z \sim \mathcal{N}(\mu_c = -1, \sigma_c^2 = 4^2)$. As $\mu_a > \mu_b, \mu_c$, class A (rightmost peak) will be the predicted class. The pairwise confidences are $P(X > Y) = 0.9831$ and $P(X > Z) = 0.8874$ (so although $\mu_c < \mu_b$, it is more likely to confuse A with C than A with B, due to the large σ_c). The lower bound is $P(X > Y) \cdot P(X > Z) = 0.8723$, whereas the exact value is $P(X > Y, Z) = 0.8740 \geq 0.8723$, as expected.

For three Gaussians (see Figure 4), we are interested in $P(X > Y, Z) = P(X > \max(Y, Z)) = \mathbb{E}_{\varphi_a}[\Phi_b(X)\Phi_c(X)]$, for which there exists no closed-form solution. However, given the pairwise probabilities $P(X > Y)$ and $P(X > Z)$, their product is a lower bound, i.e., $\mathbb{E}_{\varphi_a}[\Phi_b(X)\Phi_c(X)] \geq \mathbb{E}_{\varphi_a}[\Phi_b(X)] \cdot \mathbb{E}_{\varphi_a}[\Phi_c(X)]$, as stated in Equation 4 of the main paper and shown in the following.

In general, if f and g are strictly monotonically increasing functions over their full domain, then for a random variable X, the covariance of $f(X)$ and $g(X)$ will be non-negative. This is intuitively clear, since due to the monotonicity of the functions and their inverses, increasing or decreasing $f(X)$ will imply increasing or decreasing $g(X)$.

To prove, for the covariance of two random variables Y, Z, it generally holds that

$$\begin{aligned}\operatorname{cov}(Y, Z) &\triangleq \mathbb{E}[(Y - \mathbb{E}[Y]) \cdot (Z - \mathbb{E}[Z])] \\ &= \mathbb{E}[(Y - \mathbb{E}[Y]) \cdot Z] \\ &= \mathbb{E}[(Y - \mathbb{E}[Y]) \cdot (Z - a)]\end{aligned}$$

for any constant a. Setting $Y = f(X)$, $Z = g(X)$, and $a = g(f^{-1}(\mathbb{E}[f(X)]))$, we get:

$$\begin{aligned}&\operatorname{cov}(f(X), g(X)) \\ &\triangleq \mathbb{E}\left[(f(X) - E[f(X)]) \cdot (g(X) - \mathbb{E}[g(X)])\right] \\ &= \mathbb{E}\left[(f(X) - \mathbb{E}[f(X)]) \cdot (g(X) - g(f^{-1}(\mathbb{E}[f(X)])))\right].\end{aligned}$$

Since $f(X)$ is strictly monotonically increasing, the first term $f(X) - \mathbb{E}[f(X)]$ is positive if $X > f^{-1}(\mathbb{E}[f(X)])$ (due to strict monotonicity, $f^{-1}(y)$ is unique), and by construction, this also holds for the second term $g(X) - g(f^{-1}(\mathbb{E}[f(X)]))$, so that their product is always non-negative. It follows that $\operatorname{cov}(f(X), g(X)) \geq 0$ and thus since in general, $\operatorname{cov}(Y, Z) = \mathbb{E}[YZ] - \mathbb{E}[Y]\mathbb{E}[Z]$, it follows that $\mathbb{E}[YZ] \geq \mathbb{E}[Y]E[Z]$.

As $\Phi(x)$ are Gaussian CDFs, they are strictly monotonically increasing over the domain $x \in (-\inf, +\inf)$, and thus by setting $f(x) = \Phi_b(x)$, $g(x) = \Phi_c(x)$ and taking the expectation over φ_a, we obtain the claimed result $\mathbb{E}_{\varphi_a}[\Phi_b(X)\Phi_c(X)] \geq \mathbb{E}_{\varphi_a}[\Phi_b(X)] \cdot \mathbb{E}_{\varphi_a}[\Phi_c(X)]$.

For more than three Gaussians, the result is obtained recursively, noting that the product of any two Gaussian CDFs is also strictly monotonically increasing (which follows from the product rule and $\Phi(x) > 0$ for all $x \in \mathbb{R}$).

Please note that the fact that we are computing $P(X > Y, Z)$ does not imply any limitation of our algorithm. Especially, there is no similarity to cases where a 'one versus all' approach is used instead of 'all versus all', leading to sub-optimal results (as in support vector machines). In our case, the winner class is determined by having the largest μ, and we are only interested in the confidence associated with picking this winner class, which subsequently results in pairwise computations.

A.2 Uncertainty-aware LiDAR semantic segmentation: qualitative analysis

This section presents qualitative evaluations of uncertainty maps generated by our sampling-free method combined with deep ensembles—the configuration achieving the lowest calibration error while maintaining competitive mIoU and fast inference. Overall, we observe high uncertainty not only at misclassified points but also along class boundaries (e.g., sidewalk–street transitions), beneath vehicles—where it is often ambiguous whether to label regions as vehicle or ground, even in manual annotations—around tree trunks, and in distant areas where LiDAR measurements become sparse and noisy. These observations demonstrate the effectiveness of our method in producing reliable uncertainty estimates, which are critical for safety-sensitive applications such as autonomous driving.

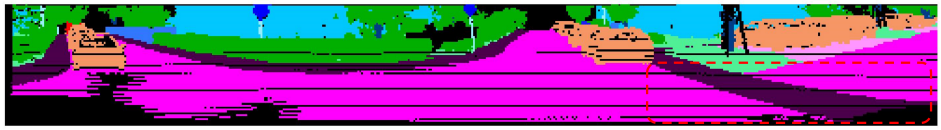
Figure 5 presents the predicted segmentation, corresponding uncertainty map, and error map for a representative LiDAR scan. A region enclosed by a dashed red box—labeled as sidewalk in the ground truth—is ambiguously classified as street, likely because it is also accessible to vehicles in this area, as seen in the camera image (Figure 5e). However, for safety reasons, it is important to distinguish this shared area between pedestrians and cars from the normal street. This semantic ambiguity is effectively captured by our method, which assigns high uncertainty in this region. In contrast, the model calibrated with temperature scaling fails to reflect this ambiguity, assigning no uncertainty despite the misclassification (Figure 5f).

Another example, shown in Figure 6, highlights a common challenge in autonomous driving: accurately identifying parking areas. In this case, the model exhibits uncertainty among three classes—road, terrain, and parking—and assigns high uncertainty to the region, effectively signaling that the classification in this area is unreliable.

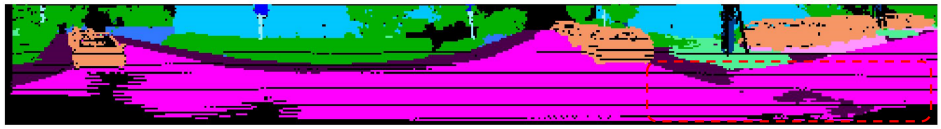
Figure 7 highlights an uncertain region in the dashed red box that appears in the camera image(7d) as a mixture of fence and vegetation, but is labeled solely as vegetation in the ground truth (7c). Our proposed approach (7a) correctly identifies this area as highly uncertain, reflecting the semantic ambiguity. In contrast, the temperature scaling method (7b) assigns low uncertainty to most of the region, with only a few isolated points marked as uncertain, despite the overall unreliability of the classification. This example also shows classes with low uncertainty, such as street, cars and sidewalk, which are classified correctly and achieving low uncertainty prediction.

A.3 Evaluation of our approach on classification tasks

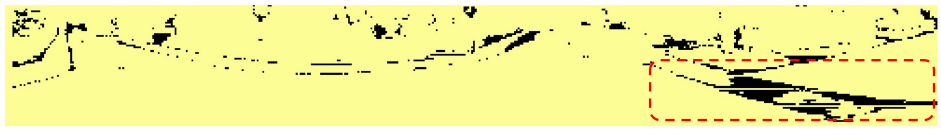
We evaluated our experiments on CIFAR-10 and CIFAR-100 [48] as benchmark datasets for classification, each containing $60,000$ $[32 \times 32 \times 3]$ color images, with CIFAR-10 divided into 10 classes and CIFAR-100 into 100 classes, providing a robust test of our approach with a varying number of classes. For these classification tasks, we utilized VGG-16 [49] and Wide-ResNet-28-10 architectures [49].



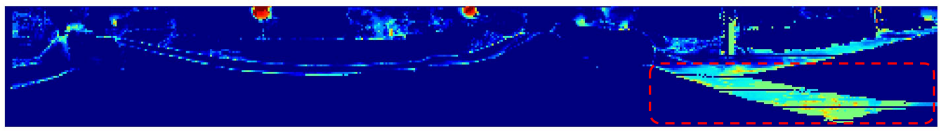
(a) Ground truth.



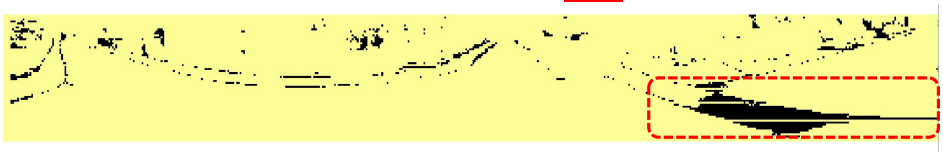
(b) Our prediction.



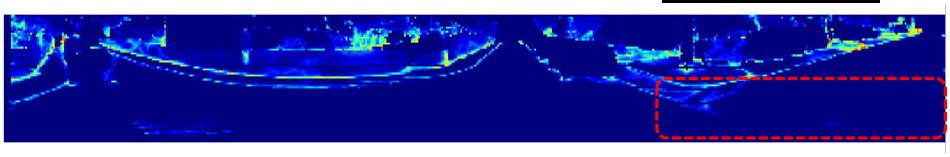
(c) Our error map. correct classifications , wrong classifications .



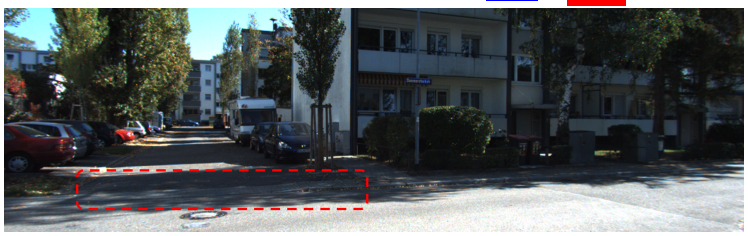
(d) Our uncertainty map, from low to high uncertainty.



(e) Error map for temperature scaling. correct classifications , wrong classifications .

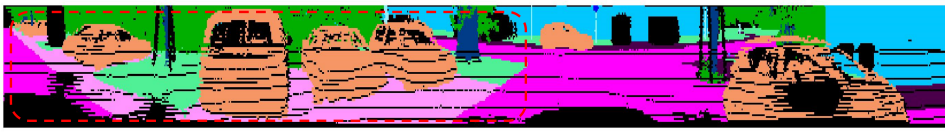


(f) Uncertainty map for temperature scaling, from low to high uncertainty.



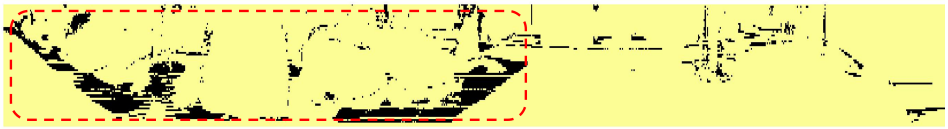
(g) Camera image.

Figure 5: Example of significant uncertainty arising from the confusion between the sidewalk and the street. The misclassified region (dashed red box) is labeled as a sidewalk in the ground truth but is also traversed by vehicles, causing overlapping classifications of street and sidewalk, which results in high uncertainty. Classes are represented with corresponding colors: outlier , parking , car , road , sidewalk , building , fence , vegetation , trunk , terrain , pole , traffic-sign .

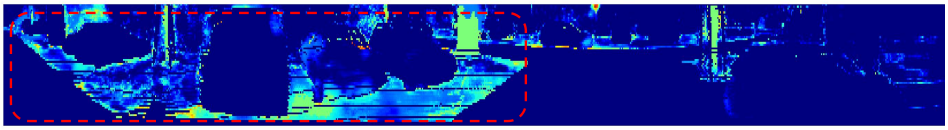


(a) Ground truth.

(b) Prediction.



(c) Error map. correct classifications , wrong classifications .



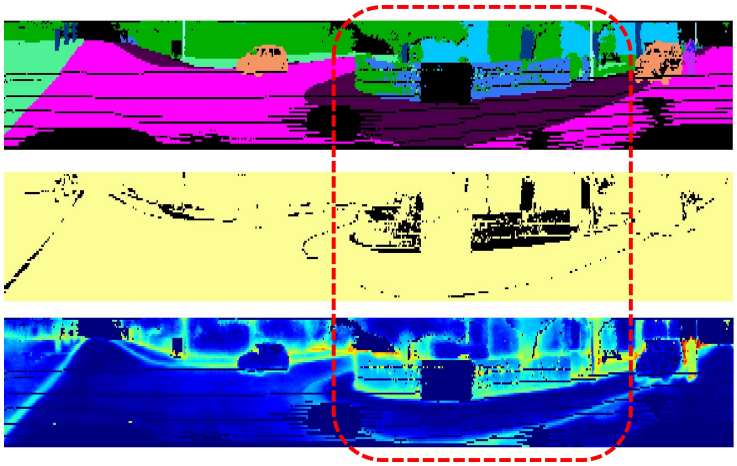
(d) Uncertainty map, from low to high uncertainty.

(e) Camera image.

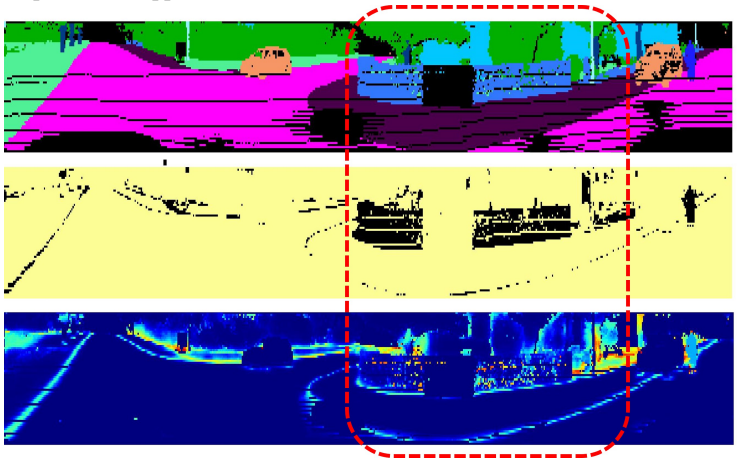
Figure 6: Example of a misclassified ground region with high uncertainty. Uncertainty map highlights classification ambiguity, showing high uncertainty in regions with unclear ground class (e.g., terrain, sidewalk, or parking) due to overlapping Gaussian distributions. Classes are represented with corresponding colors: outlier , parking , car , road , sidewalk , building , fence , vegetation , trunk , terrain , pole , traffic-sign .

Table 3: Comparative analysis of inference time, Accuracy, and ACE across various confidence calibration approaches for classification tasks on CIFAR-10 and CIFAR-100 using both the VGG-16 and WideResNet models.

	CIFAR-10, VGG-16			CIFAR-10, Wide-ResNet-28-10			CIFAR-100, VGG-16			CIFAR-100, Wide-ResNet-28-10		
Method	Accuracy(%) ↑	ACE (%) ↓	Time (s) ↓	Accuracy(%) ↑	ACE (%) ↓	Time (s) ↓	Accuracy(%) ↑	ACE (%) ↓	Time (s) ↓	Accuracy(%) ↑	ACE (%) ↓	Time (s) ↓
MCP	93.40	8.66	0.02	95.12	5.21	0.03	72.81	9.01	0.08	79.01	6.06	0.13
MCP + DE	94.01	6.89	0.13	95.70	3.38	0.18	75.68	7.86	0.50	80.23	5.43	0.84
MCP + MC dropout	93.47	6.71	0.21	95.87	3.30	0.23	75.01	7.40	0.73	80.74	5.21	1.20
logit-sampling (50 samples)	93.78	5.16	0.25	95.07	2.07	0.31	73.44	6.74	1.80	80.13	4.31	2.61
logit-sampling (50 samples)+DE	94.40	1.66	1.32	96.50	1.23	1.50	75.07	**1.16**	10.00	81.46	4.01	12.08
logit-sampling (50 samples)+MC dropout	93.93	1.40	2.70	96.10	1.20	2.20	76.50	1.23	10.80	81.08	3.98	12.28
Our sampling-free approach	93.10	5.76	0.03	95.60	1.91	0.06	73.44	6.98	0.24	80.10	4.09	0.37
Our sampling-free approach+DE	**94.81**	**1.21**	0.17	**96.66**	**0.81**	0.31	**77.03**	1.80	1.45	**82.16**	**3.61**	1.99
Our sampling-free approach+MC dropout	93.91	1.40	0.33	95.82	0.89	0.47	75.02	1.43	1.45	81.70	3.89	2.20
Temperature Scaling	93.40	1.60	0.03	95.12	2.40	0.05	72.81	2.08	0.10	79.01	4.68	0.18
EDL	92.02	4.30	0.06	94.70	3.66	0.11	73.40	5.26	0.10	78.50	5.01	0.23



(a) Predicted segmentation map, corresponding error map, and uncertainty map from our approach.



(b) Predicted segmentation map, corresponding error map, and uncertainty map from temperature scaling.

(c) Ground truth.

(d) Camera image.

Figure 7: Comparison of uncertainty estimates for a semantically ambiguous region appearing as a mixture of fence and vegetation in the camera image (7d) but labeled solely as vegetation in the ground truth (7c). Our approach (7a) effectively assigns high uncertainty to the entire region, while temperature scaling (7b) underestimates uncertainty, predicting only a few points with high uncertainty. Classes are represented with corresponding colors: outlier, parking, car, road, sidewalk, building, fence, vegetation, trunk, terrain, pole, traffic-sign.