On the calibration of underrepresented classes in LiDAR-based semantic segmentation

Mariella Dreissig[1,2,*], Florian Piewak[1] and Joschka Boedecker[2]

***Abstract*— The calibration of deep learning-based perception models plays a crucial role in their reliability. This work focuses on a class-wise evaluation of calibration measures to provide insights one challenge posed onto LiDAR-based semantic segmentation: classes with smaller instances and their subsequent underrepresentation in datasets. Those classes often include VRUs and are thus of particular interest for safety reasons. We evaluate the calibration of underrepresented classes in LiDAR-based semantic segmentation using a metric based on sparsification curves. Thereby we compare three semantic segmentation models of which one is specifically designed to better deal with smaller instances and underrepresented classes.**

I. Introduction

Environment perception allows an autonomous vehicle to detect and understand the behavior of other participants and enables it to adapt its own behavior accordingly. Deep learning methods took the performance in evironment perception to a new level by evaluating large amounts of data gathered by various sensors with different modalities. Besides cameras and RADAR sensors, in the past years the LiDAR sensor gained relevance in the context of environment perception for autonomous vehicles due to the added value of highly precise depth information.

Besides other perception tasks, semantic segmentation plays a crucial role in scene understanding for autonomous vehicles. The task of a semantic segmentation model is conducting a point-wise multi-class classification of LiDAR point clouds [1], [2]. Despite major advances in this field, the task of semantic segmentation comes with the challenge of handling severely imbalanced data [3], [4]. This is due to the natural distribution of spaces and objects, i.e. in a traffic scene there always will be significantly more measurements of the road or buildings than of persons or bicyclists.

These class imbalances have to be considered when developing and training a semantic segmentation model. While there have been approaches on overcoming this issue [5], [6]. In the context of autonomous driving, not only the detection of smaller instance classes is crucial but also having information about the reliability of those classifications. Ideally, the confidence should match the actual performance [7] and thus allow downstream tasks like sensor fusion or behavior planning to reliably interpret the models abilities. Thus, the effect of class imbalances on the calibration of a model is of particular interest regarding the safety of autonomous vehicles.

This work focuses on the analysis of underderrepresented classes in terms of calibration in unmodified and probabilistic LiDAR-based semantic segmentation models. Our contributions can be summarized as follows:

- Design of a suitable calibration metric for semantic segmentation models
- Analysis of model calibration given different confidence measures
- Comparison of three semantic segmentation models and their probabilistic versions in terms of class-wise calibration on LiDAR point clouds

II. Related Work

A. Semantic Segmentation of LiDAR point clouds

In the last years, various approaches on the semantic segmentation of point clouds have been proposed. Some operate in 3D space by utilizing voxels [8], [9] or unordered point clouds [10], [11], [12]. Other approaches project the point cloud into the 2D space in order use Convolutional Neural Networks developed for the camera-domain for the semantic segmentation of e.g. range view images [6], [13], [14], [15], [16].

Independently of the sensor modality, class imbalance is a problem which needs to be addressed in the semantic segmentation. Common ways to deal with it during training is to weight the loss function in favor of underrepresented classes [5], [17] or to construct an architecture which is able to overcome the issues imposed by smaller instances [6], [18]. In terms of performance evaluation, the well-established mean Intersection over Union (mIoU) evaluation metric accounts for class imbalances and ensures that underrepresented classes are taken into account appropriately.

B. Calibration of Deep Learning Models

In [7], Guo et al. investigate the calibration of the traditional softmax probability and proved in a series of experiments its tendency towards overconfidence. This inspired a new field of research - uncertainty estimation in deep learning [19]. Ensembling techniques like Monte-Carlo Dropout (MCD) [20] or Deep Ensembles [21] are commonly used to approximate the unknown posterior of the model weights and are known to capture the model uncertainty well. Additionally, Kendall et al. [22] introduced a technique to use probabilistic logits to learn the data uncertainty directly from the input.

Acknowledgement: This publication was compiled as part of the research project "KI Delta Learning" (project number: 19A19013A) funded by the Federal Ministry for Economic Affairs and Energy (BMWi) based on a resolution of the German Bundestag.

[1]Mercedes-Benz AG, [2]University of Freiburg, *Primary contact: mariella.dreissig@mercedes-benz.com

Works like [23] and [20] argue that the softmax probabilities are less reliable even in models with additional uncertainty estimation techniques and propose using the entropy over the softmax probability distribution as uncertainty estimates. Yet, the raw entropy is not a probability and thus cannot be converted directly into a confidence measure. This results in the majority of works using the (calibrated) softmax as confidence measure instead [24], [25], [26], [27], [28].

C. Evaluation of Calibration Measures

Guo et al. [7] proposed the Expected Calibration Error to assess the calibration of a given neural network, which is roughly speaking the correlation between the confidence and the accuracy of a model. While this produces an absolute measure and works well for tasks which actually use the accuracy as the performance metric, this approach tends to overestimate the performance for underrepresented classes. Nixon et al. [29] adapt this approach to multiclass settings, which in theory make it possible to calculate this metric for semantic segmentation tasks and weighting out the class-imbalances in the final score, but not solving the initial problem of the class-imabalances.

Mukhoti et al. [23] suggested a metric to capture particular parts of the calibration abilities of a semantic segmentation model. Their developed technique evaluates each frame patch-wise, which are labeled regarding their accuracy and uncertainty based on variable thresholds. This means, that this method requires some tuning which makes it difficult to use it for benchmarking different models.

Originally designed for the optical flow task, [30] proposed a calibration metric based on sparsification curves. It follows the same idea as [7], that the confidence should coincide with the actual performance. Thus, they define the area under the sparsification error curve (AUSE) as a relative measure for a model's calibration. This has been applied to the task of semantic segmentation as well [24], [25] using the Brier score of the softmax probabilities to rate the predictive performance.

III. Methods

A. Semantic Segmentation Model and Training

We want to uncover which role the architecture choice plays in the calibration of underrepresented classes, thus we chose three very differently designed models for the semantic segmentation task: 1. a DeeplabV3+ model with a ResNet-50 encoder as backbone [13], 2. a SalsaNeXt [26] and 3. a LiLaNet, which was specifically designed for dealing with LiDAR point clouds [6]. While the former one uses residual blocks and intense pooling to reduce computational complexity, the latter one does not use pooling in order to conserve finer structures despite the low resolution.

All models work on 2D data, thus the 3D point clouds are projected onto an image plane using the method published in [31]. We use the SemanticKITTI dataset for autonomous vision tasks [3] as a database. Both models are trained with a class-weighted cross-entropy loss based on [5] to account for the class imbalances and early stopping to avoid overfitting.

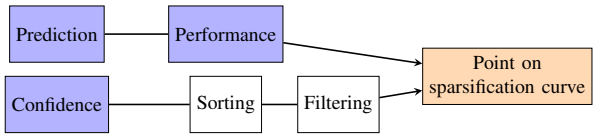

Fig. 1: *Elements of the AUSE metric.* Each point on the sparsification curve is determined by three factos (marked in blue): the prediction to determine the predictive performance on the y-axis and the confidence, which will be implicitly entered on the x-axis through ordering.

For all deterministic models we employ probabilistic versions by using MCD [20] (30 samples) and probabilistic logits [22] (5 samples) to model both aleatoric (data) and epistemic (model) uncertainty. For that, we add dropout layers (dropout rate of 0.5) in the middle part of the (encoder) model, following the findings of [32] and [23].

B. Confidence Measures

As calibration measures we evaluate both the softmax and the entropy. It has to be noted that the softmax produces a probability distribution over the classes, summing up to 1. The entropy in turn does not inherently exhibit the characteristics of a probability, but could be turned into one by normalizing with the theoretical maximum of $log(K)$. Yet, it produces only one confidence estimate conditioned on the given prediction instead of a distribution.

C. Model Calibration

Investigating the effects of underrepresented classes related to the calibration, we opted for a class-wise evaluation. Therefore, we need a metric which allows for a class-wise evaluation of any given confidence measure. At the same time, we want to decouple the calibration performance from the predictive performance as much as possible to avoid biases induced by the softmax.

Sparsification plots have been previously used to evaluate confidence measures [33]. The idea is to evaluate the point-wise confidences by creating a ranking of the confidence values. The contributing factors of the sparsification curve are illustrated in Figure 1. The performance measure only depends on the prediction and is depicted on the x-axis. The confidence measure is depicted on the y-axis implicitly by ordinal sorting. The pixels with the lowest confidence are gradually removed and the performance on the remainder of points is evaluated. If the confidence measure actually reflects the true performance, the sparsification curve should monotonically increase. Furthermore, Ilg et al. [30] suggest a normalization based on the best possible ranking according to the ground truth labels to remove the dependence on the model performance, refered to as the oracle curve. The Area Under the Sparsification Error curve (AUSE) is defined as the area under the difference between the sparsification and the oracle curve. Intuitively, the closer the sparsification curve to the ground-truth-based oracle curve is, the smaller are the AUSE values and the better is the respective calibration.

class	DeeplabV3+ deter.	DeeplabV3+ prob.	SalsaNeXt deter.	SalsaNeXt prob.	LiLaNet deter.	LiLaNet prob.
car	0.86	0.85	0.91	**0.93**	0.90	0.91
bicycle	0.00	0.01	0.04	0.05	**0.11**	**0.11**
motorcycle	0.01	0.01	0.03	0.05	**0.06**	**0.06**
truck	0.01	0.01	**0.02**	**0.02**	**0.02**	**0.02**
other-vehicle	0.04	0.05	0.08	0.10	0.11	**0.15**
person	0.05	0.07	0.17	0.19	0.20	**0.23**
bicyclist	0.03	0.08	0.12	0.17	0.15	**0.18**
motorcyclist	0.00	0.00	0.00	0.00	0.00	0.00
road	0.90	0.88	0.93	**0.94**	0.92	0.92
parking	0.06	0.06	0.10	**0.11**	0.08	0.09
sidewalk	0.72	0.69	0.77	**0.80**	0.75	0.77
other-ground	0.00	0.00	0.00	0.00	0.00	**0.01**
building	0.73	0.72	0.85	**0.86**	0.81	0.82
fence	0.11	0.13	0.22	**0.24**	0.20	0.22
vegetation	0.77	0.75	0.85	0.86	0.86	**0.87**
trunk	0.35	0.33	0.46	0.50	0.48	**0.51**
terrain	0.56	0.53	0.62	0.64	0.64	**0.65**
pole	0.35	0.33	0.58	**0.59**	0.54	0.57
traffic-sign	0.14	0.15	0.23	0.24	0.24	**0.26**
all	0.39	0.41	0.52	**0.56**	0.47	0.51

TABLE I: *IoU values for all classes and mIoU in the validation split of the SemanticKITTI dataset.* The values were calculated for both the deterministic ("deter.") and the probabilistic ("prob.") version of each model. The best performances are marked in bold.

Thus, we expect the AUSE to be inversely correlated to the predictive performance in terms of IoU.

The authors of [24] and [25] have used the Brier score using the softmax probabilities to rate the predictive performance and the entropy as confidence (uncertainty) values. This imposes two issues with our setting:

1) The Brier score requires the full probability distribution over all classes, which some confidence measures other than the softmax may not provide. Thus, it would not be possible to calculate the Brier score based on e.g. the entropy.
2) Using the entropy for the ranking (x-axis) but the softmax for the Brier score on the y-axis introduces a mixing of both measures into the metric (compare traces in Figure 1). As a result, we would not know what influences the result most: the softmax probabilities or the actual uncertainty estimation method. This makes it difficult to draw conclusions from the results in order to further improve the model.

Due to these reasons, we use the IoU for each class as measure for the predictive performance. Additionally, we investigate both the softmax probability of the argmax prediction and the entropy over the softmax as confidence measures.

IV. Results

A. Performance Evaluation

We calculate the IoU values to evaluate the predictive performance of all models and their variations. The results are listed in Table I. It is not surprising that in most cases the probabilistic versions performed better on the validation split than their deterministic counterparts. For the smaller instance classes, e.g. *bicycle*, *person* or *traffic sign*, the LiLaNet outperforms the other models due to its architecture. Contrary, the well represented classes like *car*, *road* or *building* are better learned by the SalsaNeXt model.

	DeeplabV3+ deter.	DeeplabV3+ prob.	SalsaNeXt deter.	SalsaNeXt prob.	LiLaNet deter.	LiLaNet prob.
softmax-based AUSE	1.37	1.22	1.10	**0.99**	1.05	1.15
entropy-based AUSE	1.36	1.22	1.10	1.00	1.08	1.17

TABLE II: *Overall AUSE on the validation split of the SemanticKITTI dataset for all models.* The lower the value, the better is the model calibrated.

	DeeplabV3+ deter.	DeeplabV3+ probab.	SalsaNeXt deter.	SalsaNeXt prob.	LiLaNet deter.	LiLaNet prob.
car	0.12	0.10	0.08	0.05	0.08	**0.05**
bicycle	0.53	1.32	2.13	2.12	2.15	**1.99**
motorcycle	1.71	2.08	2.77	2.87	**2.13**	2.78
truck	3.53	1.33	0.75	0.15	1.33	1.72
other-vehicle	2.69	2.75	3.00	2.89	**2.19**	2.61
person	2.48	2.06	0.92	0.82	0.81	**0.76**
bicyclist	2.56	2.56	1.42	**0.54**	1.30	1.06
motorcyclist	0.00	0.00	0.00	0.00	0.04	0.52
road	0.08	0.11	0.07	**0.04**	0.05	0.05
parking	3.32	2.75	3.25	2.74	**2.31**	2.47
sidewalk	0.58	0.70	0.48	**0.43**	0.59	0.52
other-ground	0.05	0.01	0.10	0.30	0.74	1.03
building	0.31	0.23	0.11	**0.10**	0.21	0.16
fence	2.48	1.87	**1.35**	1.91	2.13	2.19
vegetation	0.50	0.51	0.20	**0.17**	0.26	0.23
trunk	1.35	1.13	1.21	0.83	0.89	**0.77**
terrain	0.54	0.70	0.52	0.47	0.43	**0.41**
pole	1.53	1.41	**1.01**	1.14	1.63	1.35
traffic-sign	1.69	1.55	1.58	1.29	1.27	**1.18**
all (filtered)	1.44	1.32	1.26	**1.15**	**1.15**	1.16

TABLE III: *Per-class AUSE on the validation split of the SemanticKITTI dataset for all models.* The per-class AUSE is calculated for softmax probability as confidence measure. The best performances for each class are marked in bold, the underlined values indicate the best performances after filtering for outlier (marked in gray) in terms of predictive performance.

B. Calibration Evaluation with AUSE

To gain insights about the calibration of the models, we calculate the AUSE across the full validation split. This ensures that even for the underrepresented classes enough pixels are evaluated. The mean values for both confidence measures over all frames and all classes can be seen in Table II. The probabilistic SalsaNeXt achieves the best calibration, although the deterministic LiLaNet exhibits a similar AUSE value. Interestingly, in most cases the softmax probability actually outperformed the entropy in terms of calibration.

To gain deeper insights about the calibration of each class, we calculate the AUSE for all classes independently. Since the softmax performs slightly better than the entropy in terms of calibration, we focus further class-wise evaluations on the softmax. The results can be seen in III.

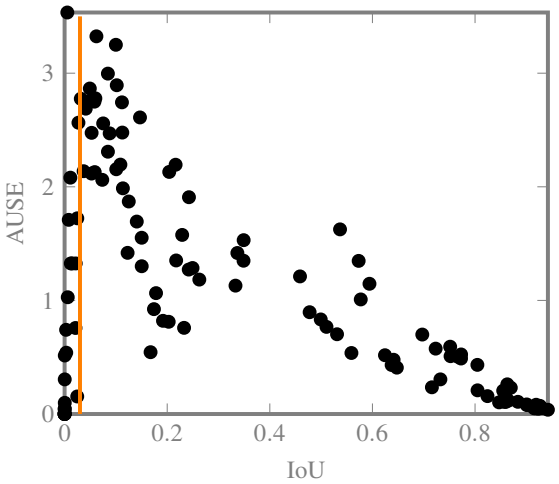

Fig. 2: *Covariation of AUSE and IoU.* An inverse linear relationship between the calibration and the predicitve performance would be desireable. In that sense, the data points left of the orange line can be seen as outliers due to a poor predictive performance.

At first glance, no meaningful pattern can be recognized from the calibration performance. When analyzing the AUSE in combination with the IoU values, the reason becomes obvious: some models were not able to learn some classes (e.g. *motorcyclist*), resulting in an IoU of 0.0. Thus, for this class the oracle as well as the sparsification curve will constantly be 0.0, resulting in a perfect calibration score. This effect is depicted in Figure 2. We expect the relation between the predictive and the calibration performance to be rougly an inverse linear correlation.

To gain some deeper insights into this phenomenon, we filter those cases (IoU < 0.03, marked with a orange line) and re-evaluate the best calibration performances. After that, a similar pattern arises as in the IoU values in Table I: the LiLaNet is better calibrated on smaller instance classes. After calculating the mean AUSE the on the filtered data, the deterministic LiLaNet exhibits the same calibration performance as the probabilistic SalsaNeXt.

V. Discussion

We demonstrated how a modification of the AUSE metric helps to analyze the calibration of a semantic segmentation model and to identify factors that contribute to the class-wise calibration. Following, we discuss our key findings based on Tables I, II and III.

1) The SalsaNeXt performed best regarding the predictive and calibration performance, followed closely by the LiLaNet: The probabilistic version of the SalsaNeXt model exhibits the best calibration performance with an AUSE value of 0.99. This value is relative, thus it is only meaningful in comparison to the values of other models. Interestingly, the next best calibration performance is achieved by the deterministic LiLaNet. This might be due to its design, which inherently produces more calibrated softmax values. It illustrates that a probabilistic model with uncertainty estimation not necessarily exhibits a better calibration. It is rather depending on the architecture, which version should be preferred. Contrary to the other two models, the DeeplabV3+ did not perform well on LiDAR data, suggesting that heavy pooling might not be beneficial when working with sparse and fine-grained structured data.

2) The difference in calibration between softmax and entropy within a model is surprisingly small: We did not observe significant difference between the softmax and the entropy with respect to their calibration abilities. This indicates, that the softmax probabilities are indeed able to capture some uncertainty in the data. Additionally, as the entropy depends on the softmax probability distribution itself, thus, when the softmax is better calibrated, it directly influences the entropy.

3) The calibration is independent of the predictive performance for a given class: Our modified AUSE metric assessed the calibration performance independently. Thus, it provides the option to create a custom weighted combination of the AUSE and the mIoU and allows for a model selection according to the desired requirements for any application.

4) The mean of the calibration metric might give misleading hints on which model actually performs best: After a closer analysis of the class-wise calibration it becomes clear, that the AUSE metric indeed is independent of the IoU metric, but contrary, the model's calibration performance is influenced by its predictive performance. That means, if a model is always wrong on one class, it does not matter whether it is well calibrated on that class or not. It will have a perfect calibration score, which does not provide any meaningful information. This effect can be avoided by filtering those classes.

VI. Conclusion and Outlook

In this paper we provided some insights about the role of class imbalance on the calibration of semantic segmentation models. We compared three models with different architectural characteristics, each in a deterministic and a probabilistic fashion. To evaluate the class-wise calibration performance, we modified sparsification-metric in order to decouple the predictive performance from the calibration. Furthermore, we gained some insights about the softmax compared to the entropy as confidence measures. Our key findings revealed that our metric is able to assess the calibration independently of the predictive performance, but in reality, the calibration and the predictive performance are influenced by each other. Furthermore, the calibration abilities depend on the structure of a model.

With this work we aim to promote research on the calibration on underrepresented classes and their effect on model performance and selection. Further research could include the evaluation of more model architectures, training strategies and uncertainty estimation methods. Additionally, it would be of interest to refine the proposed metric related to the outlier filter which influeces the calibration performance.

REFERENCES

[1] F. P. J. Piewak, "LiDAR-based Semantic Labeling (Automotive 3D Scene Understanding)," Ph.D. dissertation, Karlsruher Institut für Technologie, 2020.

[2] Y. He, H. Yu, X. Liu, Z. Yang, W. Sun, Y. Wang, Q. Fu, Y. Zou, and A. Mian, "Deep Learning based 3D Segmentation: A Survey," *arXiv*, 2021.

[3] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, J. Gall, and C. Stachniss, "Towards 3D LiDAR-based Semantic Scene Understanding of 3D Point Cloud Sequences-The SemanticKITTI," in *Int. J. Robot. Res.*, 2021.

[4] L. Triess. (2019) Scripts for SemanticKITTI Dataset Statistics. [Online]. Available: https://github.com/ltriess/semantic_kitti_stats

[5] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation," *arXiv*, 2016.

[6] F. Piewak, P. Pinggera, M. Schäfer, D. Peter, B. Schwarz, N. Schneider, M. Enzweiler, D. Pfeiffer, and M. Zöllner, "Boosting LiDAR-based semantic Labeling by Cross-Modal Training Data Deneration," in *ECCV*, 2019.

[7] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in *ICML*, 2017.

[8] J. Huang and S. You, "Point Cloud Labeling using 3D Convolutional Neural Network," in *ICPR*, 2016.

[9] H.-Y. Meng, L. Gao, Y. Lai, and D. Manocha, "VV-Net: Voxel VAE Net with Group Convolutions for Point Cloud Segmentation," in *ICCV*, 2018.

[10] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," in *NIPS*, 2017.

[11] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution On X-Transformed Points," in *NIPS*, 2018.

[12] J. Yang, Q. Zhang, B. Ni, L. Li, J. Liu, M. Zhou, and Q. Tian, "Modeling Point Clouds with Self-Attention and Gumbel Subset Sampling," in *CVPR*, 2019.

[13] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018.

[14] B. Wu, A. Wan, X. Yue, and K. Keutzer, "SqueezeSeg: Convolutional Neural Nets with Recurrent CRF for Real-Time Road-Object Segmentation from 3D LiDAR Point Cloud," in *ICRA*, 2017.

[15] 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 *ECCV*, 2020.

[16] E. E. Aksoy, S. Baci, and S. Cavdar, "SalsaNet: Fast Road and Vehicle Segmentation in LiDAR Point Clouds for Autonomous Driving," in *IV*, 2019.

[17] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "RangeNet++: Fast and Accurate LiDAR Semantic Segmentation," in *IROS*, 2019.

[18] R. Cheng, R. Razani, E. Taghavi, E. Li, and B. Liu, "(AF)2-S3Net: Attentive Feature Fusion with Adaptive Feature Selection for Sparse Semantic Segmentation Network," in *CVPR*, 2021.

[19] F. Arnez, H. Espinoza, A. Radermacher, and F. Terrier, "A Comparison of Uncertainty Estimation Approaches in Deep Learning Components for Autonomous Vehicle Applications," in *IJCAI - Workshop*, 2020.

[20] Y. Gal, "Uncertainty in Deep Learning," Ph.D. dissertation, University of Cambridge, 2016.

[21] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles," in *NIPS*, 2017.

[22] A. Kendall and Y. Gal, "What Uncertainties do we Need in Bayesian Deep Learning for Computer Vision?" in *NIPS*, 2017.

[23] J. Mukhoti and Y. Gal, "Evaluating Bayesian Deep Learning Methods for Semantic Segmentation," in *CVPR - Workshop*, 2020.

[24] Y. Shen, Z. Zhang, M. R. Sabuncu, and L. Sun, "Real-Time Uncertainty Estimation in Computer Vision via Uncertainty-Aware Distribution Distillation," in *WACV*, 2021.

[25] F. K. Gustafsson, M. Danelljan, and T. B. Schön, "Evaluating Scalable Bayesian Deep Learning Methods for Robust Computer Vision," in *CVPR - Workshop*, 2020.

[26] T. Cortinhal, G. Tzelepis, and E. E. Aksoy, "SalsaNext: Fast, Uncertainty-aware Semantic Segmentation of LiDAR Point Clouds for Autonomous Driving," in *ISVC*, 2020.

[27] S. Cygert, B. Wróblewski, K. Woźniak, R. Słowiński, and A. Czyżewski, "Closer Look at the Uncertainty Estimation in Semantic Segmentation under Distributional Shift," in *IJCNN*, 2021.

[28] T. Pearce, A. Brintrup, and J. Zhu, "Understanding Softmax Confidence and Uncertainty," *arXiv*, 2021.

[29] J. Nixon, M. Dusenberry, G. Jerfel, T. Nguyen, J. Liu, L. Zhang, and D. Tran, "Measuring Calibration in Deep Learning," *arXiv*, 2019.

[30] E. Ilg, Özgün Çiçek, S. Galesso, A. Klein, O. Makansi, F. Hutter, and T. Brox, "Uncertainty Estimates and Multi-Hypotheses Networks for Optical Flow," in *ECCV*, 2018.

[31] L. T. Triess, D. Peter, C. B. Rist, and J. M. Zöllner, "Scan-based Semantic Segmentation of LiDAR Point Clouds: An Experimental Study," in *IV*, 2020.

[32] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene Understanding," in *BMVC*, 2017.

[33] A. S. Wannenwetsch, M. Keuper, and S. Roth, "ProbFlow: Joint Optical Flow and Uncertainty Estimation," in *ICCV*, 2017.