GLENet: Boosting 3D Object Detectors with Generative Label Uncertainty Estimation

Yifan Zhang · Qijian Zhang · Zhiyu Zhu · Junhui Hou · Yixuan Yuan

Received: date / Accepted: date

Abstract The inherent ambiguity in ground-truth annotations of 3D bounding boxes, caused by occlusions, signal missing, or manual annotation errors, can confuse deep 3D object detectors during training, thus deteriorating detection accuracy. However, existing methods overlook such issues to some extent and treat the labels as deterministic. In this paper, we formulate the label uncertainty problem as the diversity of potentially plausible bounding boxes of objects. Then, we propose GLENet, a generative framework adapted from conditional variational autoencoders, to model the one-to-many relationship between a typical 3D object and its potential ground-truth bounding boxes with latent variables. The label uncertainty generated by GLENet is a plug-and-play module and can be conveniently integrated into existing deep 3D detectors to build probabilistic detectors and supervise the learning of the localization uncertainty. Besides, we propose an uncertainty-aware quality estimator architecture in probabilistic detectors to guide the training of the IoU-branch with predicted localization uncertainty. We incorporate the proposed methods into various popular base 3D detectors and demonstrate significant and consistent performance gains on both KITTI and Waymo benchmark datasets. Especially, the proposed GLENet-VR outperforms all published LiDAR-based approaches by a large margin and achieves the top rank among single-modal methods on the challenging KITTI test set. The source code and pre-trained models are publicly available at https://github.com/Eaphan/GLENet.

Keywords 3D object detection · label uncertainty · conditional variational autoencoders · probabilistic object detection · 3D point cloud

1 Introduction

As one of the most practical application scenarios of computer vision, 3D object detection has been attracting much academic and industrial attention in the current deep learning era with the rise of autonomous driving and the emergence of large-scale annotated datasets (e.g., KITTI (Geiger et al., 2012), and Waymo (Sun et al., 2020)).

In the current community, despite the proliferation of various deep learning-based 3D detection pipelines, it is observed that mainstream 3D object detectors are typically designed as deterministic models, without considering the critical issue of the ambiguity of annotated ground-truth labels. However, different aspects of ambiguity/inaccuracy inevitably exist in the ground-truth annotations of object-level bounding boxes, which may significantly influence the overall learning process of such deterministic detectors. For example, in the data collection phase, raw point clouds can be highly incomplete due to the intrinsic properties of LiDAR sensors as well as uncontrollable environmental occlusion. Moreover, in the data labeling phase, ambiguity naturally occurs when different human annotators subjectively estimate object shapes and locations from 2D images and partial 3D points. To facilitate intuitive understandings, we provide typical examples in Fig. 1, from which we can observe that an incomplete LiDAR observation can correspond to multiple potentially plausible labels, and objects with similar LiDAR observation can be annotated with significantly varying bounding boxes.

Yifan Zhang, Qijian Zhang, Zhiyu Zhu, and Junhui Hou
Department of Computer Science, City University of Hong Kong.
E-mail: {yzhang3362-c, qijizhang3-c, zhiyuzhu2-c}@my.cityu.edu.hk; jh.hou@cityu.edu.hk;

Yixuan Yuan
Department of Electronic Engineering, The Chinese University of Hong Kong.
E-mail: yxyuan@ee.cuhk.edu.hk

This project was supported by the Hong Kong Research Grants Council under Grants 11202320 and 11218121. Corresponding author: Junhui Hou

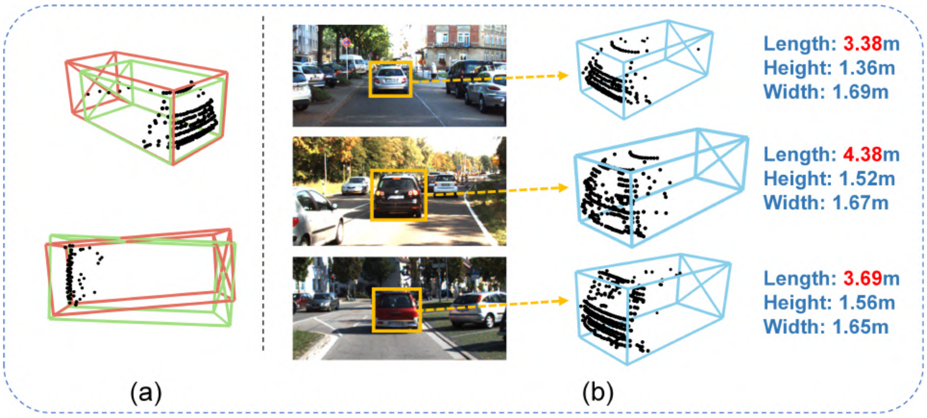


Fig. 1: (a) Given an object with an incomplete LiDAR observation, there may exist multiple potentially plausible ground-truth bounding boxes with varying sizes and shapes. (b) Ambiguity and inaccuracy can be inevitable in the labeling process when annotations are derived from 2D images and partial points. In the given cases, similar point clouds of the *car* category with only the *rear* part can be annotated with different ground-truth boxes of varying lengths.

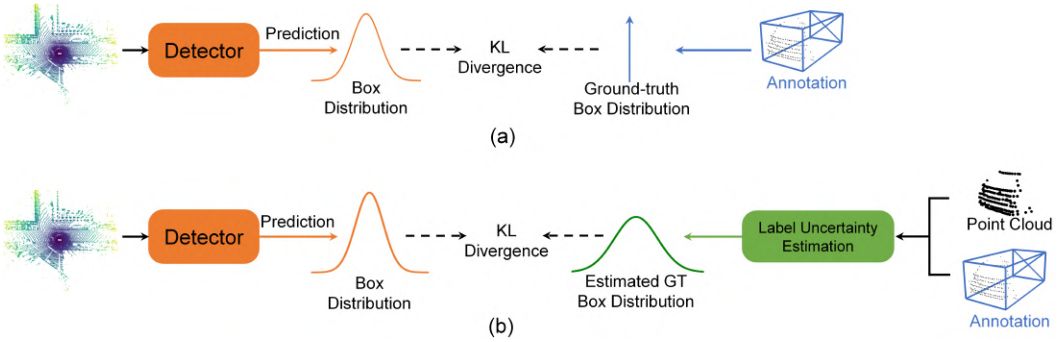


Fig. 2: Illustration of two different learning paradigms of probabilistic object detectors. (a) Methods that adopt probabilistic modeling in the detection head but essentially still ignore the issue of ambiguity in ground-truth bounding boxes. (b) Methods that explicitly estimate ground-truth bounding box distributions to be used as more reliable supervision signals.

Motivated by the aforementioned phenomena, there also exists another family of probabilistic detectors that explicitly consider the potential influence of label ambiguity. Conclusively, these methods can be categorized into two paradigms, as illustrated in Fig. 2. The first paradigm of learning frameworks (He et al., 2019; Meyer et al., 2019; Feng et al., 2018, 2019) tends to output the probabilistic distribution of bounding boxes instead of directly regressing definite box coordinates in a deterministic fashion. For example, under the pre-assumption of a Gaussian distribution, the detection head predicts the mean and variance of the distribution accordingly. To supervise such probabilistic models, these works simply treat ground-truth bounding boxes as the Dirac delta distribution, after which KL divergence is applied between the estimated distributions and ground truths. Obviously, the major limitation of these methods lies in that they fail to essentially address the problem of label ambiguity, since the ground-truth bounding boxes are still considered deterministic with zero uncertainty (i.e., modeled as a Dirac delta function). To this end, the second paradigm of learning frameworks attempts to quantify label uncertainty derived from some simple heuristics (Meyer and Thakurdesai (2020)) or Bayes (Wang et al. (2020)), such that the detectors can be supervised under a more reliable bounding box distribution. However, it is unsurprising that these approaches still cannot produce satisfactory label uncertainty estimation results due to insufficient modeling capacity. In general, this line of work is still in its initial stage with a very limited number of studies, despite its greater potential in generating higher-quality label uncertainty estimation in a data-driven manner.

Architecturally, this work follows the second type of design philosophy, where we particularly customize a powerful deep learning-based label uncertainty quantification framework to enhance the reliability of the estimated ground-truth bounding box distributions. Technically, we formulate the label uncertainty problem as the diversity of potentially plausible bounding boxes and explicitly model the one-to-many relationship between a typical 3D object and its potentially

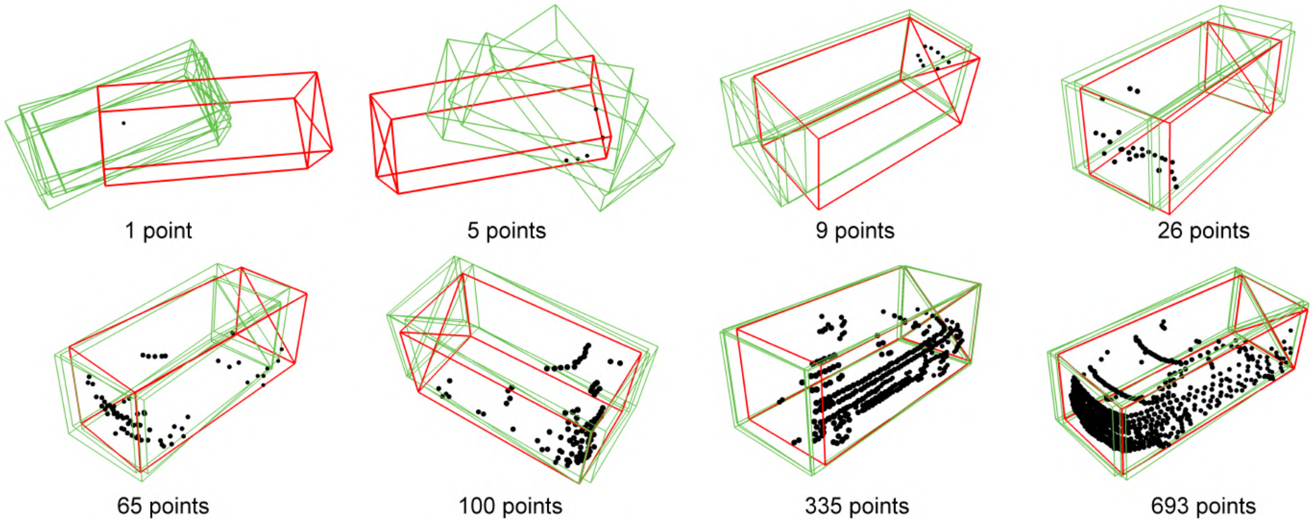


Fig. 3: Illustration of multiple potentially plausible bounding boxes from GLENet on the KITTI dataset by sampling latent variables multiple times. The point cloud, annotated ground-truth boxes, and predictions of GLENet are colored in black, red, and green, respectively. GLENet produces diverse predictions for objects represented with sparse point clouds and incomplete outlines, and consistent bounding boxes for objects with high-quality point clouds. The variance of the multiple predictions by GLENet is used to estimate the uncertainty of the annotated ground-truth bounding boxes.

plausible ground-truth boxes in a learning-based framework. We propose GLENet, a novel deep generative network adapted from conditional variational auto-encoders (CVAE), which introduces a latent variable to capture the distribution over potentially plausible bounding boxes of point cloud objects. During inference, we sample latent variables multiple times to generate diverse bounding boxes (see Fig.3), the variance of which is taken as label uncertainty to guide the learning of localization uncertainty estimation in the downstream detection task. Besides, based on the observation that detection results with low localization uncertainty in probabilistic detectors tend to have accurate actual localization quality (see Section4.2), we further propose the uncertainty-aware quality estimator (UAQE), which facilitates the training of the IoU-branch with the localization uncertainty estimation.

To demonstrate our effectiveness and universality, we integrate GLENet into several popular 3D object detection frameworks to build powerful probabilistic detectors. Experiments on KITTI (Geiger et al., 2012) and Waymo (Sun et al., 2020) datasets demonstrate that our method can bring consistent performance gains and achieve the current state-of-the-art. Particularly, the proposed GLENet-VR surpasses all published single-modal detection methods by a large margin and **ranks** 1^{st} among all published LiDAR-based approaches on the highly competitive KITTI 3D detection benchmark on March 29^{th}, 2022[1].

We summarize the main contributions of this paper as follows:

- We are the first to formulate the 3D label uncertainty problem as the diversity of potentially plausible bounding boxes of objects. To capture the one-to-many relationship between a typical 3D object and its potentially plausible ground-truth bounding boxes, we present a deep generative model named GLENet. Additionally, we introduce a general and unified deep learning-based paradigm, including the network structure, loss function, evaluation metric, etc.
- Inspired by the strong correlation between the localization quality and the predicted uncertainty in probabilistic detectors, we propose UAQE to facilitate the training of the IoU-branch.

The remainder of the paper is organized as follows. Section 2 reviews existing works on LiDAR-based detectors and label uncertainty estimation methods. In Section 3, we explicitly formulate the label uncertainty estimation problem from the probabilistic distribution perspective, followed by the technical implementation of GLENet. In Section 4, we introduce a unified way of integrating the label uncertainty statistics predicted by GLENet into the existing 3D object detection frameworks to build more powerful probabilistic detectors, as well as some theoretical analysis. In Section 5, we conduct experiments on the KITTI dataset and the Waymo Open dataset to demonstrate the effectiveness of our method in enhancing existing 3D detectors and the ablation study to analyze the effect of different components. Finally, Section 7 concludes this paper.

[1] www.cvlibs.net/datasets/kitti/eval_object.php?obj_benchmark=3d

2 Related Work

2.1 LiDAR-based 3D Object Detection

Existing 3D object detectors can be classified into two categories: single-stage and two-stage. For single-stage detectors, Zhou and Tuzel (2018) proposed to convert raw point clouds to regular volumetric representations and adopted voxel-based feature encoding. Yan et al. (2018b) presented a more efficient sparse convolution. Lang et al. (2019) converted point clouds to sparse fake images using pillars. Shi and Rajkumar (2020a) aggregated point information via a graph structure. He et al. (2020) introduced point segmentation and center estimation as auxiliary tasks in the training phase to enhance model capacity. Zheng et al. (2021a) constructed an SSFA module for robust feature extraction and a multi-task head for confidence rectification, and proposed DI-NMS for post-processing. For two-stage detectors, Shi et al. (2020b) exploited a voxel-based network to learn the additional spatial relationship between intra-object parts under the supervision of 3D box annotations. Shi et al. (2019) proposed to directly generate 3D proposals from raw point clouds in a bottom-up manner, using semantic segmentation to validate points to regress detection boxes. The follow-up work (Yang et al., 2019) further proposed PointsPool to convert sparse proposal features to compact representations and used spherical anchors to generate accurate proposals. Shi et al. (2020a) utilized both point-based and voxel-based methods to fuse multi-scale voxel and point features. Deng et al. (2021) proposed voxel RoI pooling to extract RoI features from coarse voxels.

To address the boundary ambiguity problems in 3D object detection caused by occlusion and signal miss, some studies, such as SPG (Xu et al., 2021), have tried to use point cloud completion methods to restore the full shape of objects and improve the detection performance (Yan et al., 2021; Najibi et al., 2020). However, generating complete and precise shapes with incomplete point clouds remains a non-trivial task.

2.2 Probabilistic 3D Object Detector

There are two types of uncertainty in deep learning predictions. A type of uncertainty called aleatoric uncertainty is caused by the inherent noise in observational data, which cannot be eliminated. The other type is called epistemic Uncertainty or model uncertainty, which is caused by incomplete training and can be alleviated with more training data. Most existing state-of-the-art 2D (Liu et al., 2016; Tan et al., 2020; Carion et al., 2020) and 3D (Shi et al., 2020b) object detectors produce a deterministic box with a confidence score for each detection. While the probability score represents the existence and semantic confidence, it cannot reflect the uncertainty about predicted localization well. By contrast, probabilistic object detectors (He et al., 2019; Harakeh et al., 2020; Varamesh and Tuytelaars, 2020) estimate the probabilistic distribution of predicted bounding boxes rather than take them as deterministic results. For example, He et al. (2019) and Choi et al. (2019) modeled the predicted boxes as Gaussian distributions, the variance of which can indicate the localization uncertainty and is predicted with additional layers in the detection head. It introduces the KL Loss between the predicted Gaussian distribution and the ground-truth bounding boxes modeled as a Dirac delta function, so the regression branch is expected to output a larger variance and get a smaller loss for inaccurate localization estimation for the cases with ambiguous boundaries. Li et al. (2021) facilitated the learning of localization quality with distribution statistics of a bounding box, such as the mean value, which inspires us to further utilize the estimated uncertainty in UAQE. Meyer et al. (2019) proposed a probabilistic 3D object detector modeling the distribution of bounding box corners as a Laplacian distribution.

However, most probabilistic detectors take the ground-truth bounding box as a deterministic Dirac delta distribution and ignore the ambiguity in the ground-truth. Therefore, the localization variance is actually learned in an unsupervised manner, which may result in sub-optimal localization precision and erratic training (see our theoretical analysis in Section 4.1).

2.3 Label Uncertainty Estimation

Label noise (or uncertainty) is a common problem in real-world datasets and could seriously affect the performance of supervised learning algorithms. As the neural network is prone to overfit to even complete random noise (Zhang et al. (2021)), it is important to prevent the network from overfitting noisy labels. An obvious solution is to consider the label of a misclassified sample to be uncertain and remove the samples (Delany et al., 2012). Garcia et al. (2015) used a soft voting approach to approximate a noise level for each sample based on the aggregation of the noise degree prediction calculated for a set of binary classifiers. Luengo et al. (2018) extended this work by correcting the label when most classifiers predict the same label for noisy samples. Confident Learning Northcutt et al. (2021) estimated uncertainty in dataset labels by estimating the joint distribution of noisy labels and true labels. However, the above studies mainly focus on the image classification task.

There only exists a limited number of previous works focusing on quantifying uncertainty statistics of annotated ground-truth bounding boxes. Meyer and Thakurdesai (2020) proposed to model label uncertainty by the IoU between the label bounding box and the corresponding convex hull of the aggregated LiDAR observations. However, it is non-learning-based and thus has limited modeling capacity. Besides, it only produces uncertainty of the ground-truth box as a whole instead of each dimension. Wang et al. (2020) proposed a Bayes method to estimate label noises by quantifying the matching degree of point clouds for the given boundary box with the Gaussian Mixture Model. However, its assumption of conditional probabilistic independence between point clouds is often untenable in practice. Differently, we formulate label uncertainty as the diversity of potentially plausible bounding boxes. There may be some objects with few points that exactly match the learned surface points of the corresponding labeled bounding box, so the label is considered by (Wang et al. (2020)) to be deterministic. But for an object with sparse point clouds, our GLENet will output different and plausible bounding boxes and further estimate high label uncertainty based on them, regardless of whether points match the given label. In general, Wang et al. (2020) used the Bayesian paradigm to estimate the correctness of the annotated box as the label uncertainty, while our method formulates it as the diversity of potentially plausible bounding boxes and predicts it by GLENet.

2.4 Conditional Variational Auto-Encoder

The variational auto-encoder (VAE) (Kingma and Welling, 2014) has been widely used in image and shape generation tasks (Yan et al., 2016; Nash and Williams, 2017). It transforms natural samples into a distribution where latent variables can be drawn and passed to a decoder network to generate diverse samples. Sohn et al. (2015) introduced the conditional variational autoencoder (CVAE), which extends the capabilities of the traditional VAE by incorporating an additional condition during the generative process. The CVAE model consists of an encoder, a decoder, and an extra input, which is usually a label or other structured information pertinent to the generation task. This auxiliary condition enables the CVAE to generate more targeted and controlled samples compared to its unsupervised counterpart, the VAE. In the NLP field, VAE has been widely applied to many text generation tasks, such as dialogue response (Zhao et al., 2017), machine translation (Zhang et al., 2016), story generation (Wang and Wan, 2019), and poem composing (Li et al., 2018). VAE and CVAE have also been applied in computer vision tasks, like image generation (Yan et al., 2016), human pose estimation (Sharma et al., 2019), medical image segmentation (Painchaud et al., 2020), salient object detection (Li et al., 2019; Zhang et al., 2020), and modeling human motion dynamics (Yan et al., 2018a). Recently, VAE and CVAE algorithms have also been applied extensively to applications of 3D point clouds, such as generating grasp poses (Mousavian et al., 2019) and instance segmentation (Yi et al., 2019).

Inspired by CVAE for generating diverse reasonable responses in dialogue systems, we propose GLENet adapted from CVAE to capture the one-to-many relationship between objects with incomplete point clouds and the potentially plausible ground-truth bounding boxes. To the best of our knowledge, we are the first to employ CVAE in 3D object detection to model label uncertainty.

3 Proposed Label Uncertainty Estimation

As aforementioned, the ambiguity of annotated ground-truth labels widely exists in 3D object detection scenarios and has adverse effects on the deep model learning process, which is not well addressed or even completely ignored by previous works. To this end, we propose GLENet, a generic and unified deep learning framework that generates label uncertainty by modeling the one-to-many relationship between point cloud objects and potentially plausible bounding box labels. Then the variance of the multiple outputs of GLENet for a single object is computed as the label uncertainty, which is extended as an auxiliary regression objective to enhance the performance of the downstream 3D object detection task.

3.1 Problem Formulation

Let $C = \{c_i\}_{i=1}^n$ be a set of n observed LiDAR points belonging to an object, where $c_i \in \mathbb{R}^3$ is a 3D point represented with spatial coordinates. Let X be the annotated ground-truth bounding box of C parameterized by the center location (c_x, c_y, c_z), the size (length l , width w, and height h), and the orientation r, i.e., $X = [c_x,\ c_y,\ c_z,\ w,\ l,\ h,\ r] \in \mathbb{R}^7$.

We formulate the uncertainty of the annotated ground-truth label of an object as the diversity of potentially plausible bounding boxes of the object, which could be quantitatively measured with the variance of the distribution of the potential bounding boxes. First, we model the distribution of these potential boxes conditioned on point cloud C, denoted as $p(X|C)$. Specifically, based on the Bayes theorem, we introduce an intermediate variable z to write the conditional distribution as

$$p(X|C) = \int_z p(X|z, C)p(z|C)dz. \tag{1}$$

Then, with $p(X|z, C)$ and $p(z|C)$ known, we can adopt a Monte Carlo method to get multiple bounding box predictions by sampling z multiple times and approximate the variance of $p(X|C)$ with that of the sampled predictions.

In the following, we will introduce our learning-based framework named GLENet to realize the estimation process.

3.2 Inference Process of GLENet

Fig. 4 (a) shows the flowchart of GLENet parameterized by neural parameters θ, which aims to predict $p(z|C)$ and $p(X|z, C)$. Specifically, under the assumption that the prior

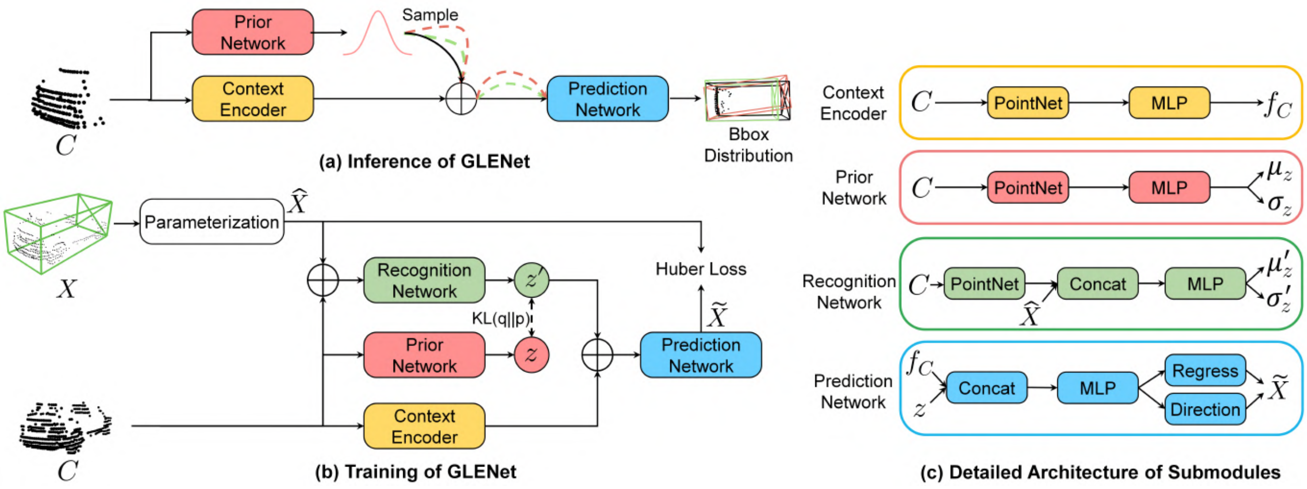


Fig. 4: The overall workflow of GLENet. In the training phase, we learn parameters μ and σ (resp. μ' and σ') of latent variable z (resp. z') through the prior network (resp. recognition network), after which a sample of z' and the corresponding geometrical embedding produced by the context encoder are jointly exploited to estimate the bounding box distribution. In the inference phase, we sample from the distribution of z multiple times to generate different bounding boxes, whose variance we use as label uncertainty. Note we denote multiple sampling with black, orange, and green lines in subgraph (a).

distribution $p(z|C)$ subjects to a multivariate Gaussian distribution parameterized by (μ_z, σ_z), denoted as $\mathcal{N}(\mu_z, \sigma_z^2)$, we design a prior network, which is composed of PointNet (Qi et al., 2017) and additional MLP layers, from the input point cloud C to predict the values of (μ_z, σ_z). Then, we employ a context encoder to embed the input point cloud C into a high dimensional feature space, leading to the geometric feature representation f_C, which is concatenated with z sampled from $\mathcal{N}(\mu_z, \sigma_z^2)$ and fed into a prediction network composed of MLPs to regress the bounding box distribution $p(X|z, C)$, i.e., the localization, dimension, and orientation of the bounding box.

As empirically observed in various related domains (Goyal et al., 2017), it could be difficult to make use of latent variables when the prediction network can generate a plausible output only using the sufficiently expressive features of condition C. Therefore, we utilize a simplified PointNet architecture as the backbone of the context encoder to avoid posterior collapse. We refer the readers to Section 5.1.3 for the implementation details of these modules. In the following sections, we also use $p_\theta(z|C)$, $p_\theta(X|z, C)$, and $p_\theta(X|C)$ to denote the predictions of $p(z|C)$, $p(X|z, C)$, and $p(X|C)$ by GLENet, respectively.

3.3 Training Process of GLENet

3.3.1 Recognition Network

Given C and its annotated bounding box X, we assume there is a true posterior distribution $q(z|X, C)$. Thus, during training, we construct a recognition network parameterized by network parameters ϕ (see Fig. 4 (b)) to learn an auxiliary posterior distribution $q_\phi(z'|X, C)$ subjecting to a Gaussian distribution, denoted as $\mathcal{N}(\mu'_z, \sigma_z'^2)$, to regularize $p_\theta(z|C)$, i.e., $p_\theta(z|C)$ should be close to $q_\phi(z'|X, C)$.

Specifically, for the recognition network, we adopt the same learning architecture as the prior network to generate point cloud embeddings, which are concatenated with ground-truth bounding box information and fed into the subsequent MLP layers to learn $q_\phi(z'|X, C)$. Moreover, to facilitate the learning process, we encode the information X into offsets relative to predefined anchors, and then perform normalization as:

$$\begin{aligned} &t_{c_x} = \frac{c_x^{gt}}{d^a},\ t_{c_y} = \frac{c_y^{gt}}{d^a},\ t_{c_z} = \frac{c_z^{gt}}{h^a}, \\ &t_w = \log\frac{w^{gt}}{w^a},\ t_l = \log\frac{l^{gt}}{l^a},\ t_h = \log\frac{h^{gt}}{h^a}, \\ &t_r = \sin(r^{gt}), \end{aligned} \tag{2}$$

where (w^a, l^a, h^a) is the size of the predefined anchor located in the center of the point cloud, and $d^a = \sqrt{(l^a)^2 + (w^a)^2}$ is the diagonal of the anchor box. We also take $\cos(r)$ as the additional input of the recognition network to handle the issue of angle periodicity.

3.3.2 Objective Function

Following CVAE Sohn et al. (2015), we optimize GLENet by maximizing the variational lower bound of the conditional log-likelihood $p_\theta(X|C)$:

$$\begin{aligned} \log p_\theta(X|C) \ \geq\ & E_{q_\phi(z'|X,C)}[\log p_\theta(X|z, C)] - \\ & KL(q_\phi(z'|X, C)||p_\theta(z|C)), \end{aligned} \tag{3}$$

where $E_q[p]$ returns the expectation of p on the distribution of q, and $KL(\cdot)$ denotes KL-divergence.

Specifically, the first term $E_{q_\phi(z'|X,C)}[\log p_\theta(X|z,C)]$ enforces the prediction network to be able to restore ground-truth bounding box from latent variables. Following (Yan et al. (2018b)) and (Deng et al. (2021)), we explicitly define the bounding box reconstruction loss as

$$L_{rec} = L_{rec}^{reg} + \lambda L_{rec}^{dir}, \tag{4}$$

where L_{rec}^{reg} denotes the Huber loss imposed on the prediction and encoded regression targets as described in Eq. (2), and L_{rec}^{dir} denotes the binary cross-entropy loss used for direction classification.

The second term $KL(q_\phi(z'|X,C)\|p_\theta(z|C))$ is aimed at regularizing the distribution of z by minimizing the KL-divergence between $p_\theta(z|C)$ and $q_\phi(z'|X,C)$. Since $p_\theta(z|C)$ and $q_\phi(z'|X,\ C)$ are re-parameterized as $\mathcal{N}(\mu_z,\sigma_z^2)$ and $\mathcal{N}(\mu_z',\ \sigma_z'^2)$ through the prior network and the recognition network, respectively, we can explicitly define the regularization loss as:

$$L_{KL}(q_\phi(z'|X,C)\|p_\theta(z|C)) = \log\frac{\sigma_z'}{\sigma_z} + \frac{\sigma_z^2}{2\sigma_z'^2} + \frac{(\mu_z-\mu_z')^2}{2\sigma_z'^2}. \tag{5}$$

Thus, the overall objective function is written as

$$L = L_{rec} + \gamma L_{KL}, \tag{6}$$

where we empirically set the hyperparameter γ to 1 in all experiments.

4 Probabilistic 3D Detectors with Label Uncertainty

To reform a typical detector to be a probabilistic object detector, we can enforce the detection head to estimate a probability distribution over bounding boxes, denoted as $P_\Theta(y)$, instead of a deterministic bounding box location:

$$P_\Theta(y) = \frac{1}{\sqrt{2\pi\hat{\sigma}^2}} e^{-\frac{(y-\hat{y})^2}{2\hat{\sigma}^2}}, \tag{7}$$

where Θ indicates learnable network weights of a typical detector, $\hat{y}$ is the predicted bounding box location, and $\hat{\sigma}$ is the predicted localization variance.

Accordingly, we also assume the ground-truth bounding box as a Gaussian distribution $P_D(y)$ with variance σ^2, whose value is estimated by GLENet:

$$P_D(y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-y_g)^2}{2\sigma^2}}, \tag{8}$$

where y_g represents the ground-truth bounding box. Therefore, we can incorporate the generated label uncertainty in the KL loss between the distribution of prediction and ground-truth in the detection head:

$$\begin{aligned} L_{reg} &= D_{KL}(P_D(y)\|P_\Theta(y)) \\ &= \log\frac{\hat{\sigma}}{\sigma} + \frac{\sigma^2}{2\hat{\sigma}^2} + \frac{(y_g-\hat{y})^2}{2\hat{\sigma}^2}. \end{aligned} \tag{9}$$

4.1 More Analysis of KL-Loss

When ignoring label ambiguity and formulating the ground-truth bounding box as a Dirac delta function, as done in (He et al. (2019)), the loss in Eq. (9) degenerates into

$$L_{reg}^{prob} \propto \frac{\log(\hat{\sigma}^2)}{2} + \frac{(y_g-\hat{y})^2}{2\hat{\sigma}^2}, \tag{10}$$

and the partial derivative of Eq. (10) with respect to the predicted variance $\hat{\sigma}$ is:

$$\frac{\partial L_{reg}^{prob}}{\partial\hat{\sigma}} = \frac{1}{\hat{\sigma}} - \frac{(y_g-\hat{y})^2}{\hat{\sigma}^3}. \tag{11}$$

When minimizing Eq. (10), a potential issue is that as $|y_g - \hat{y}| \to 0$,

$$\frac{\partial L_{reg}^{prob}}{\partial\hat{\sigma}} \to \frac{1}{\hat{\sigma}}, \tag{12}$$

the derivative for $\hat{\sigma}$ can explode when $\hat{\sigma} \to 0$. Based on the property of KL-loss, the prediction is optimal only when the estimated $\hat{\sigma} = 0$ and the localization error $|y_g - \hat{y}| = 0$. Therefore, the gradient explosion may result in erratic training and sub-optimal localization precision.

By contrast, after modeling the ground-truth bounding box as a Gaussian distribution, the partial derivative of Eq. (9) with respect to prediction is:

$$\frac{\partial L_{reg}}{\partial\hat{\sigma}} = \frac{1}{\hat{\sigma}} - \frac{\sigma^2}{\hat{\sigma}^3} - \frac{(y_g-\hat{y})^2}{\hat{\sigma}^3}, \tag{13}$$

and

$$\frac{\partial L_{reg}}{\partial\hat{y}} = \frac{\hat{y}-y_g}{\hat{\sigma}^2}. \tag{14}$$

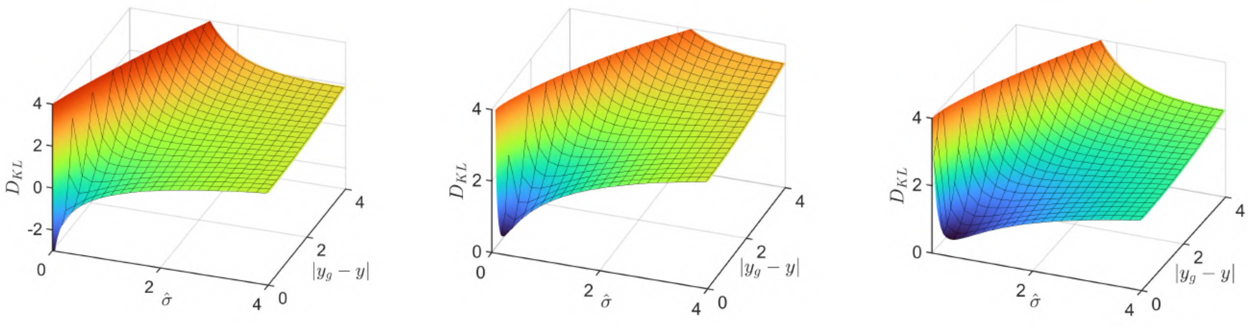
As $|y_g - \hat{y}| \to 0$ and $\hat{\sigma} > 0$,

$$\frac{\partial L_{reg}}{\partial\hat{\sigma}} \to \frac{1}{\hat{\sigma}}(1 - \frac{\sigma^2}{\hat{\sigma}^2}), \tag{15}$$

and

$$\frac{\partial L_{reg}}{\partial\hat{y}} \to 0. \tag{16}$$

Thus, when the predicted distribution reaches the optimal solution that is the distribution of ground-truth, i.e., $|y_g-\hat{y}| \to 0$ and $\hat{\sigma} \to \sigma$, the derivatives for both $\hat{y}$ and $\hat{\sigma}$ become zero, which is an ideal property for the loss function and avoids the aforementioned gradient explosion issue.



(a) $L_{reg}^{prob}\,(\sigma = 0)$ (b) $L_{reg}\,(\sigma = 0.2)$ (c) $L_{reg}\,(\sigma = 0.5)$

Fig. 5: Illustration of the KL-divergence between distributions as a function of localization error $|y_g - \hat{y}|$ and estimated localization variance $\hat{\sigma}$ given different label uncertainty σ. With label uncertainty σ estimated by GLENet instead of zero, the gradient is smoother when the loss converges to the minimum. Besides, the L_{reg} is smaller when σ is larger, which prevents the model from overfitting to uncertain annotations.

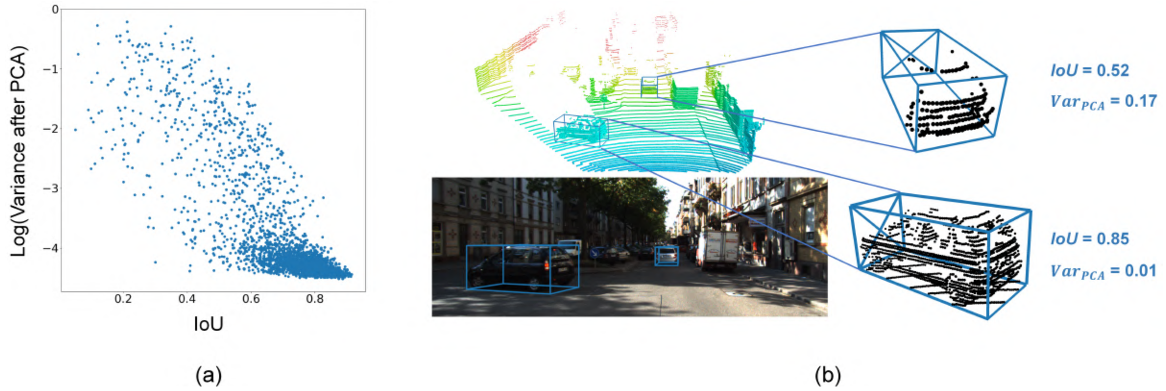


Fig. 6: (a) Illustration of the relationship between the actual localization precision (i.e., IoU between predicted and ground-truth bounding box) and the variance predicted by a probabilistic detector. Here, we reduce the dimension of the variance with PCA to facilitate visualization. (b) Two examples: for the sparse sample, the prediction has high uncertainty and low localization quality, while for the dense sample, the prediction has high localization quality and low uncertainty estimation.

Fig. 5 shows the landscape of the KL-divergence loss function under different label uncertainty σ, which are markedly different in shape and property. The L_{reg}^{prob} approaches infinitesimal and the gradient explodes as $|y_g - \hat{y}| \to 0$ and $\hat{\sigma} \to 0$. However, when we introduce the estimated label uncertainty and the predicted distribution is equal to the ground-truth distribution, the KL Loss has a determined minimum value of 0.5 and the gradient is smoother.

4.2 Uncertainty-aware Quality Estimator

Most state-of-the-art two-stage 3D object detectors use an IoU-related confidence score as the sorting criterion in NMS (non-maximum suppression), indicating the localization quality rather than the classification score. As shown in Fig. 6, there is a strong correlation between the uncertainty and actual localization quality for each bounding box. This observation motivates us to use uncertainty as a criterion for judging the quality of the boxes. However, the estimated uncertainty is 7-dimensional, making it infeasible to directly replace the IoU confidence score with the uncertainty. To overcome this

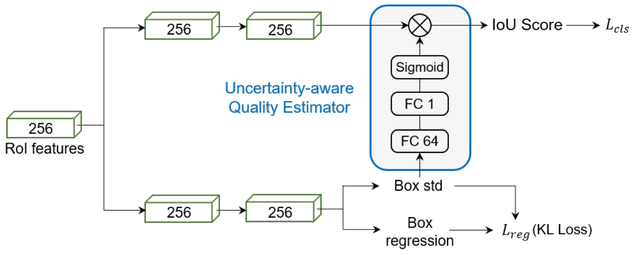


Fig. 7: Illustration of the proposed UAQE module in the detection head using the learned localization variance to assist the training of localization quality (IoU) estimation branch.

Algorithm 1: 3D Variance Voting

```
Data: $B$ is an $N \times 7$ matrix of predicted bounding boxes with
      parameters $(x, y, z, w, l, h, \theta)$. $C$ is the
      corresponding variance. $S$ is a set of N corresponding
      confidence values. $\sigma_t$ is a tunable hyperparameter.
Result: The final voting results $D$ of selected candidate boxes.
$B = \{b_1, b_2, ..., b_N\}$; and $C = \{c_1, c_2, ..., c_N\}$;
$S = \{s_1, s_2, ..., s_N\}$; and $L = \{1, 2, ..., N\}$;
$D \leftarrow \{\}$;
$iou_{thresh} \leftarrow \mu$;
while $L \neq \emptyset$ do
    idx $=\underset{i \in L}{\mathrm{argmax}}\, S, b' = b_{idx}$;
    $L' = \{i | i \in L, IoU(b_i, b') > iou_{thresh}\}$;
    $P \leftarrow \{\}$;
    for $i \in L'$ do
        $p_i = e^{-(1-IoU(b_i,b))^2/\sigma_t}$;
        if $|tan(b_i^\theta - b'^\theta)| > 1$ then
            $p_i^\theta = 0$;
        end
        $P \leftarrow P \bigcup p_i$;
    end
    $b_m = \frac{\sum_{i \in L'} b_i \cdot p_i / c_i}{\sum_{i \in L'} p_i / c_i}, p_i \in P, b_i \in B, c_i \in C$;
    $D \leftarrow D \bigcup b_m$;
    $L \leftarrow L - L'$;
end
```

issue, we propose an uncertainty-aware quality estimator (UAQE) that introduces uncertainty information to facilitate the training of the IoU-branch and improve the accuracy of IoU estimation. The UAQE is shown in Fig. 7. Given the predicted uncertainty as input, we construct a lightweight sub-module consisting of two fully connected (FC) layers followed by the Sigmoid activation to generate a coefficient. The original output of the IoU-branch is then multiplied with this coefficient to obtain the final estimation. The UAQE aims to capture the uncertainty in the estimation and adjust the final output accordingly, resulting in a more accurate estimation of the IoU score.

4.3 3D Variance Voting

Considering that in probabilistic object detectors, the learned localization variance by the KL loss can reflect the uncertainty of the predicted bounding boxes, following (He et al., 2019), we also propose 3D variance voting to combine neighboring bounding boxes to seek a more precise box representation. Specifically, at a single iteration in the loop, box b with the maximum score is selected and its new location is calculated according to itself and the neighboring boxes. During the merging process, the neighboring boxes that are closer and have a low variance are assigned higher weights. Note that neighboring boxes with a large angle difference from b do not participate in the ensembling of angles. We refer the readers to Algorithm 1 for the details.

5 Experiments

To reveal the effectiveness and universality of our method, we integrated GLENet into several popular types of 3D object detection frameworks to form probabilistic detectors, which were evaluated on two commonly used benchmark datasets, i.e., the Waymo Open dataset (WOD) (Sun et al., 2020) and the KITTI dataset (Geiger et al., 2012). Specifically, we start by introducing specific experiment settings and implementation details in Section 5.1. After that, we report the detection performance of the resulting probabilistic detectors and make comparisons with previous state-of-the-art approaches in Sections 5.2 and 5.3. Finally, we conduct a series of ablation studies to verify the necessity of different key components and configurations in Section 5.4.

5.1 Experiment Settings

5.1.1 Benchmark Datasets

The **KITTI** dataset contains 7481 training samples with annotations in the camera field of vision and 7518 testing samples. According to the occlusion level, visibility, and bounding box size, the samples are further divided into three difficulty levels: simple, moderate, and hard. Following common practice, when performing experiments on the val set, we further split all training samples into a subset with 3712 samples for training and the remaining 3769 samples for validation. We report the performance on both the val set and online test leaderboard for comparison. And we use all training data for the test server submission.

The **Waymo Open** dataset is a large-scale autonomous driving dataset with more diverse scenes and object annotations in full 360°, which contains 798 sequences (158361 LiDAR frames) for training and 202 sequences (40077 LiDAR frames) for validation. These frames are further divided into two difficulty levels: LEVEL1 for boxes with more than five points and LEVEL2 for boxes with at least one point. We report performance on both LEVEL 1 and LEVEL 2 difficulty objects using the recommended metrics, mean Average Precision (mAP) and mean Average Precision weighted by heading accuracy (mAPH). To conduct the experiments efficiently, we created a representative training set by randomly selecting 20% of the frames from the original training set, which comprises approximately 32,000 frames. All evaluations were performed on the complete validation set, consisting of around 40,000 frames, using the official evaluation tool.

5.1.2 Evaluation Metric for GLENet

Due to the unavailability of the true distribution of a ground-truth bounding box, we propose to evaluate GLENet in a

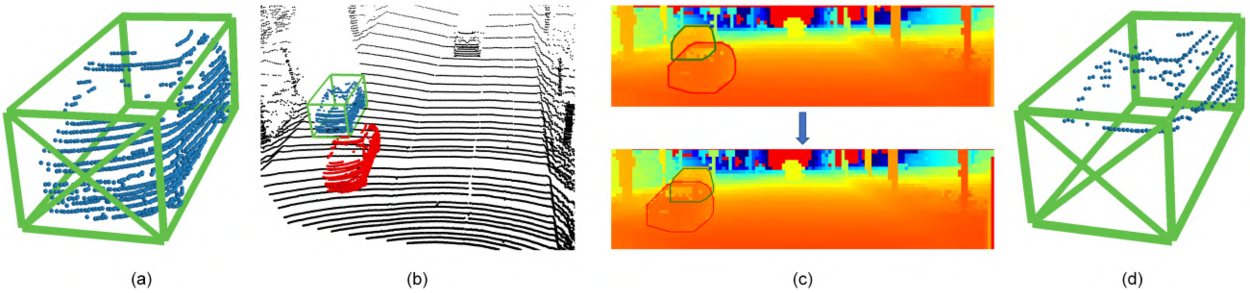


Fig. 8: Illustration of the occlusion data augmentation. (a) The point cloud of the original object associated with the annotated ground-truth bounding box. (b) A sampled dense object (red) is placed between the LiDAR sensor and the original object (blue). (c) The projected range image from the point cloud in (b), where the convex hull (the red polygon) of the sampled object is calculated and further jittered to increase the diversity of occluded samples. Based on the convex hull (the green polygon) of the original point cloud, the occluded area can be obtained. The point cloud of the original object corresponding to the occluded area is removed. (d) Final augmented object with the annotated ground-truth bounding boxes.

non-reference manner, in which the negative log-likelihood between the estimated distribution of ground-truth $p_D(X|C)$ subjecting to a Gaussian distribution $\mathcal{N}(\hat{t}, \sigma^2)$ and $p_\theta(X|C)$ is computed:

$$\begin{aligned} L_{NLL}(\theta) &= -\int p_\theta(X|C) \log p_D(X|C) \mathrm{d}X \\ &\approx -\frac{1}{S}\sum_{i=1}^{S} \log p_D(X_i|C) \\ &= -\frac{1}{S}\sum_{i=1}^{S} \sum_{\substack{k\in\{c_x,c_y,\\ c_z,w,l,h,r\}}} \frac{(t_k^i - \hat{t}_k^i)^2}{2{\sigma_k}^2} + \frac{\log(\sigma_k^2)}{2} + \frac{\log(2\pi)}{2}, \end{aligned} \tag{17}$$

where S denotes the number of inference times, X_i is the result of the i-th inference, and $\hat{t}_k^i$ and t_k^i represent the regression targets and the predicted offsets, respectively. We estimate the integral by randomly sampling multiple prediction results via the Monte Carlo method. Generally, the value of L_{NLL} is small when GLENet outputs reasonable bounding boxes, i.e., predicting diverse plausible boxes with high variance for incomplete point cloud and consistent, precise boxes with low variance for high-quality point cloud, respectively.

5.1.3 Implementation Details

To prevent data leakage, we kept the dataset division of GLENet consistent with that of the downstream detectors. As the initial input of GLENet, the point cloud of each object was uniformly pre-processed into 512 points via random subsampling/upsampling. Then we decentralized the point cloud by subtracting the coordinates of the center point to eliminate the local impact of translation.

Architecturally, we realized the prior network and recognition network with an identical PointNet structure consisting of three FC layers of output dimensions (64, 128, 512), followed by another FC layer to generate an 8-dim latent variable. To avoid posterior collapse, we particularly chose a lightweight PointNet structure with channel dimensions (8, 8, 8) in the context encoder. The prediction network concatenates the generated latent variable and context features and feeds them into subsequent FC layers of channels (64, 64) before predicting offsets and directions.

5.1.4 Training and Inference Strategies

To optimize GLENet, we adopted Adam (Kingma and Ba, 2015) with a learning rate of 0.003, β_1 of 0.9, and β_2 of 0.99. The model was trained for a total of 400 epochs on KITTI and 40 epochs on Waymo, with a batch size of 64 on 2 GPUs. We used the one-cycle policy (Smith, 2017) to update the learning rate.

In the training process, we applied common data augmentation strategies, including random flipping, scaling, and rotation, in which the scaling factor and rotation angle were uniformly drawn from [0.95, 1.05] and $[-\pi/4, \pi/4]$, respectively. It is important to include multiple plausible ground-truth boxes in training, especially for incomplete point clouds, so we further propose an occlusion-driven augmentation approach, as illustrated in Fig. 8, after which a complete point cloud may look similar to another incomplete point cloud, while the ground-truth boxes of them are completely different. To overcome posterior collapse, we also adopted KL annealing (Bowman et al., 2016) to gradually increase the weight of the KL loss from 0 to 1. We followed k-fold cross-sampling to divide all training objects into ten mutually exclusive subsets. To overcome overfitting, each time we trained GLENet on 9 subsets and then made predictions on the remaining subset to generate label uncertainty estimations on the whole training set. During inference, we sampled the latent variable z from the predicted prior distribution $p_\theta(z|c)$ 30 times to form multiple predictions, the variance of which was used as the label uncertainty.

Table 1: Quantitative comparison with state-of-the-art methods on the KITTI test set for vehicle detection, under the evaluation metric of 3D Average Precision (AP) of 40 sampling recall points. The best and second-best results are highlighted in bold and underlined, respectively.

Method	Reference	Modality	3D AP$_{R40}$			
			Easy	Mod.	Hard	mAP
MV3D (Chen et al., 2017)	CVPR'17	RGB+LiDAR	74.97	63.63	54.00	64.20
F-PointNet (Qi et al., 2018)	CVPR'18	RGB+LiDAR	82.19	69.79	60.59	70.86
MMF (Liang et al., 2019)	CVPR'19	RGB+LiDAR	88.40	77.43	70.22	78.68
PointPainting (Vora et al., 2020)	CVPR'20	RGB+LiDAR	82.11	71.70	67.08	73.63
CLOCs (Pang et al., 2020)	IROS'20	RGB+LiDAR	88.94	80.67	77.15	82.25
EPNet (Huang et al., 2020)	ECCV'20	RGB+LiDAR	89.81	79.28	74.59	81.23
3D-CVF (Yoo et al., 2020)	ECCV'20	RGB+LiDAR	89.20	80.05	73.11	80.79
STD (Yang et al., 2019)	ICCV'19	LiDAR	87.95	79.71	75.09	80.92
Part-A2 (Shi et al., 2020b)	TPAMI'20	LiDAR	87.81	78.49	73.51	79.94
3DSSD (Yang et al., 2020)	CVPR'20	LiDAR	88.36	79.57	74.55	80.83
SA-SSD (He et al., 2020)	CVPR'20	LiDAR	88.80	79.52	72.30	80.21
PV-RCNN (Shi et al., 2020a)	CVPR'20	LiDAR	90.25	81.43	76.82	82.83
PointGNN (Shi and Rajkumar, 2020b)	CVPR' 20	LiDAR	88.33	79.47	72.29	80.03
Voxel-RCNN (Deng et al., 2021)	AAAI'21	LiDAR	90.90	81.62	77.06	83.19
SE-SSD (Zheng et al., 2021b)	CVPR'21	LiDAR	91.49	82.54	77.15	83.73
VoTR (Mao et al., 2021b)	ICCV'21	LiDAR	89.90	82.09	**79.14**	83.71
Pyramid-PV (Mao et al., 2021a)	ICCV'21	LiDAR	88.39	82.08	77.49	82.65
CT3D (Sheng et al., 2021)	ICCV'21	LiDAR	87.83	81.77	77.16	82.25
GLENet-VR (Ours)	-	LiDAR	**91.67**	**83.23**	78.43	**84.44**

Table 2: Quantitative comparison of different methods on the KITTI validation set for vehicle detection, under the evaluation metric of 3D Average Precision (AP) calculated with 11 sampling recall positions. The 3D APs under 40 recall sampling recall points are also reported for the moderate car class. The best and second-best results are highlighted in bold and underlined, respectively.

Methods	Reference	3D AP$_{R11}$			3D AP$_{R40}$		
		Easy	Moderate	Hard	Easy	Moderate	Hard
Part-A^2 (Shi et al., 2020b)	TPAMI'20	89.47	79.47	78.54	-	-	-
3DSSD (Yang et al., 2020)	CVPR'20	89.71	79.45	78.67	-	-	-
SA-SSD (He et al., 2020)	CVPR'20	90.15	79.91	78.78	92.23	84.30	81.36
PV-RCNN (Shi et al., 2020a)	CVPR'20	89.35	83.69	78.70	92.57	84.83	83.31
SE-SSD (Zheng et al., 2021b)	CVPR'21	**90.21**	85.71	**79.22**	93.19	**86.12**	83.31
VoTR (Mao et al., 2021b)	ICCV'21	89.04	84.04	78.68	-	-	-
Pyramid-PV (Mao et al., 2021a)	ICCV'21	89.37	84.38	78.84	-	-	-
CT3D (Sheng et al., 2021)	ICCV'21	89.54	86.06	78.99	92.85	85.82	83.46
SECOND (Yan et al., 2018b)	Sensors'18	88.61	78.62	77.22	91.16	81.99	78.82
GLENet-S (Ours)	-	88.68	82.95	78.19	91.73	84.11	81.35
CIA-SSD (Zheng et al., 2021a)	AAAI'21	90.04	79.81	78.80	**93.59**	84.16	81.20
GLENet-C (Ours)	-	89.82	84.59	78.78	93.20	85.16	81.94
Voxel R-CNN (Deng et al., 2021)	AAAI'21	89.41	84.52	78.93	92.38	85.29	82.86
GLENet-VR (Ours)	-	89.93	**86.46**	79.19	93.51	86.10	**83.60**

5.1.5 Base Detectors

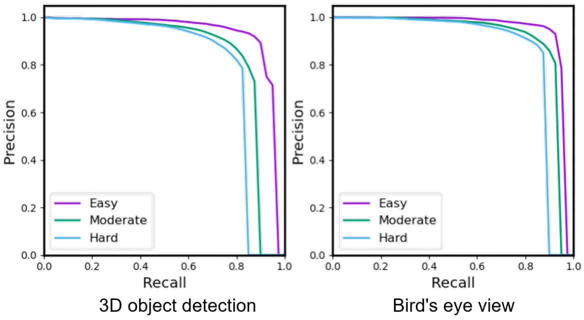
We integrated GLENet into four popular deep 3D object detection frameworks, i.e., SECOND (Yan et al., 2018b), CIA-SSD (Zheng et al., 2021a), CenterPoint (two-stage) (Yin et al., 2021), and Voxel R-CNN (Deng et al., 2021), to construct probabilistic detectors, which are dubbed as GLENet-S, GLENet-C, GLENet-CP, and GLENet-VR, respectively. Specifically, we introduced an extra FC layer on the top of the detection head to estimate standard deviations along with the box locations. Meanwhile, we applied the proposed UAQE to GLENet-VR to facilitate the training of the IoU-branch. Generally, we set the value of σ_t to 0.05 and the value of μ to 0.01 in KITTI and 0.7 in Waymo dataset in 3D variance voting. Note that for fair comparisons, we kept the network configurations of these base detectors unchanged except those related to the new submodules.

5.2 Evaluation on the KITTI Dataset

We compared GLENet-VR with state-of-the-art detectors on the KITTI test set, and Table 1 reports the AP and mAP that averages over the APs of easy, moderate and hard objects. As

Table 3: Performance comparisons on the KITTI val set for pedestrian and cyclist class using AP_{R11}.

Method	Pedestrian			Cyclist		
	Easy	Moderate	Hard	Easy	Moderate	Hard
Second	56.55	**52.97**	47.73	80.59	67.14	63.11
GLENet-S	**58.22**	52.39	**49.53**	**82.67**	**68.29**	**65.62**
Voxel R-CNN	**66.32**	60.52	55.42	86.62	70.69	66.05
GLENet-VR	66.18	**62.05**	**56.00**	**87.28**	**74.07**	**70.90**



3D object detection

Bird's eye view

Fig. 9: PR curves of GLENet-VR on the car class of the KITTI test set.

of March 29^{th}, 2022, our GLENet-VR surpasses all published single-modal detection methods by a large margin and ranks 1^{st} among all published LiDAR-based approaches. Besides, Fig. 9 also provides the detailed Precision-Recall (PR) curves of GLENet-VR on KITTI test split.

Table 2 lists the validation results of different detection frameworks on the KITTI dataset, from which we can observe that GLENet-S, GLENet-C, and GLENet-VR consistently outperform their corresponding baseline methods, i.e., SECOND, CIA-SSD, and Voxel R-CNN, by 4.79%, 4.78%, and 1.84% in terms of 3D R11 AP on the category of moderate car. Particularly, GLENet-VR achieves 86.36% AP on the moderate car class, which surpasses all other state-of-the-art methods. Besides, as a single-stage method, GLENet-C achieves 84.59% AP for the moderate vehicle class, which is comparable to the existing two-stage approaches while achieving relatively lower inference costs. It is worth noting that our method is compatible with mainstream detectors and can be expected to achieve better performance when combined with stronger base detectors. Besides, our method also performs well on other classes. As shown in Table 3, GLENet-S outperforms the Second by +1.8% and +2.51% on pedestrian and cyclist classes respectively for 3D AP on the hard difficulty. And for the baseline Voxel R-CNN, our method improves the performance by +1.47% and +3.38% on pedestrian and cyclist classes respectively on the moderate difficulty.

5.3 Evaluation on the Waymo Open Dataset

In Table 4, we present a comprehensive comparison of various state-of-the-art methods for vehicle detection on the Waymo Open Dataset, considering both LEVEL_1 and LEVEL_2 difficulty settings. The evaluation metrics used in this comparison include the 3D mean Average Precision (mAP) for different distance ranges (0-30m, 30-50m, and 50m-inf) and the overall mAP for LEVEL_1 and LEVEL_2. Specifically, our method contributes 2.44%, 1.21% and 1.24% improvements in terms of LEVEL_1 mAP for SECOND, CenterPoint-TS and Voxel R-CNN, respectively. The improvements observed in the table demonstrate that our method is robust and consistently enhances the performance of baseline models like SECOND and Voxel R-CNN. And GLENet-VR demonstrates the best performance with an mAP of 77.32% and 69.68% for LEVEL 1 and LEVEL 2, respectively. This superior performance can be attributed to the effective handling of bounding box am-

Table 4: Quantitative comparison of different methods on the Waymo validation set for vehicle detection. ⋆: experiment results re-produced with the code of OpenPCDet[a]. The best and second-best results are highlighted in bold and underlined, respectively.

Methods	LEVEL_1 3D mAP				mAPH	LEVEL_2 3D mAP				mAPH
	Overall	0-30m	30-50m	50m-inf	Overall	Overall	0-30m	30-50m	50m-inf	Overall
PointPillar (Lang et al., 2019)	56.62	81.01	51.75	27.94	-	-	-	-	-	-
MVF (Zhou et al., 2020)	62.93	86.30	60.02	36.02	-	-	-	-	-	-
PV-RCNN (Shi et al., 2020a)	70.30	91.92	69.21	42.17	69.69	65.36	91.58	65.13	36.46	64.79
VoTr-TSD (Mao et al., 2021b)	74.95	92.28	73.36	51.09	74.25	65.91	-	-	-	65.29
Pyramid-PV (Mao et al., 2021a)	76.30	92.67	74.91	54.54	75.68	67.23	-	-	-	66.68
CT3D (Sheng et al., 2021)	76.30	92.51	75.07	55.36	-	69.04	91.76	68.93	42.60	-
SECOND⋆ (Yan et al., 2018b)	69.85	90.71	68.93	41.17	69.40	62.76	86.92	62.57	35.89	62.30
GLENet-S (Ours)	72.29	91.02	71.86	45.43	71.85	64.78	87.56	65.11	38.60	64.25
CenterPoint-TS⋆ (Yin et al., 2021)	75.52	92.09	74.35	54.27	75.07	67.37	90.89	68.11	42.46	66.94
GLENet-CP (Ours)	76.73	92.70	75.70	55.77	76.27	68.50	91.95	69.43	43.68	68.08
Voxel R-CNN⋆ (Deng et al., 2021)	76.08	92.44	74.67	54.69	75.67	68.06	91.56	69.62	42.80	67.64
GLENet-VR (Ours)	**77.32**	**92.97**	**76.28**	**55.98**	**76.85**	**69.68**	**92.09**	**71.21**	**44.36**	**68.97**

[a] Reference: https://github.com/open-mmlab/OpenPCDet.

biguity, especially for distant and sparse point cloud objects. In addition to the overall performance, our methods also exhibit noteworthy improvements in the 30-50m and 50m-inf distance ranges. These results indicate that our method is particularly effective in resolving ambiguity for objects that are farther away from the sensor, which has traditionally posed challenges for point cloud-based detection algorithms. In conclusion, Table 4 highlights the superior performance of our methods in 3D detection tasks on the Waymo Open Dataset. By effectively addressing the challenges posed by distant and sparse point cloud objects, our method demonstrates significant improvements in both LEVEL_1 and LEVEL_2 difficulty settings across various distance ranges.

5.4 Ablation Study

We conducted ablative analyses to verify the effectiveness and characteristics of our processing pipeline. In this section, all the involved model variants are built upon the Voxel R-CNN baseline and evaluated on the KITTI dataset, under the evaluation metric of average precision calculated with 40 recall positions.

5.4.1 Comparison with Other Label Uncertainty Estimation

We compared GLENet with two other ways of label uncertainty estimation: 1) treating the label distribution as the deterministic Dirac delta distribution with zero uncertainty; 2) estimating the label uncertainty with simple heuristics, i.e., the number of points in the ground-truth bounding box or the IoU between the label bounding box and its convex hull of the aggregated LiDAR observations (Meyer and Thakurdesai, 2020). As shown in Table 5, our method consistently outperforms existing label uncertainty estimation paradigms. Compared with heuristic strategies, our deep generative learning paradigm can adaptively estimate label uncertainty statistics in 7 dimensions, instead of the uncertainty of bounding boxes as a whole, considering the variance in each dimension could be very different.

Besides, to compare with (Wang et al., 2020), whose code is not publicly available, we evaluated our method under its experiment settings and compared results with its reported performance. As shown in Table 6, our method outperforms (Wang et al., 2020) significantly in terms of AP$_{BEV}$ on both moderate and hard levels.

5.4.2 Key Components of Probabilistic Detectors

We analyzed the contributions of different key components in our constructed probabilistic detectors and reported results in Table 7. According to the second row, we can conclude that only training with the KL loss brings little performance gain. Introducing the label uncertainty generated by GLENet into the KL Loss contributes 0.75%, 0.51%, and 0.3% improvements on the APs of easy, moderate, and hard classes, respectively, which demonstrates its regularization effect on KL-loss (Eq. 9) and its ability to estimate more reliable uncertainty statistics of bounding box labels. The proposed UAQE module in the probabilistic detection head boosts the easy, moderate, and hard APs by 0.25%, 0.19% and 0.15%, respectively, validating its effectiveness in estimating the localization quality.

Table 5: Comparison of different label uncertainty estimation approaches. "Convex hull" refers to the method in (Meyer and Thakurdesai, 2020). The best results are highlighted in bold.

Methods	3D AP$_{R40}$		
	Easy	Moderate	Hard
Voxel R-CNN	92.38	85.29	82.86
GLENet-VR w/ L_{KLD} (σ^2=0)	92.48	85.37	83.05
GLENet-VR w/ L_{KLD} (points num)	92.46	85.58	83.16
GLENet-VR w/ L_{KLD} (convex hull)	92.33	85.45	82.81
GLENet-VR w/ L_{KLD} (Ours)	**93.49**	**86.10**	**83.56**

Table 6: Comparison of our method with (Wang et al., 2020) on the KITTI val set. The best results are highlighted in bold.

Method	AP_{BEV} for IoU@0.7		
	Easy	Mod.	Hard
PIXOR (Yang et al., 2018)	86.79	80.75	76.60
ProbPIXOR + $\mathcal{L}_{KLD}$ ($\sigma = 0$)	88.60	80.44	78.74
ProbPIXOR + $\mathcal{L}_{KLD}$ (Wang et al., 2020)	**92.22**	82.03	79.16
ProbPIXOR + $\mathcal{L}_{KLD}$ (Ours)	91.50	**84.23**	**81.85**

To gain a better understanding of how UAQE enhances the estimation of IoU-related confidence scores (the location quality), we analyze the error in IoU estimation for both GLENet-VR and the baseline model (w/o UAQE) over different actual IoU values between the proposals and their corresponding ground-truth boxes. Figure 10 illustrates the changes in the error distribution of IoU estimation. We can observe that the UAQE module effectively reduces the IoU estimation error across various intervals of actual IoU values, such as [0.1, 0.6). These findings demonstrate that the UAQE module not only improves the overall average precision (AP) metric but also enhances the accuracy of location quality estimation.

5.4.3 Ablation Study of GLENet

Effectiveness of Preprocessing. As mentioned previously, to eliminate the local impact of translation on the input of GLENet, the point cloud of a single object is standardized to zero mean value. However, this process might remove meaningful information contained in distances. For instance, distant objects with fewer points typically have high label uncertainty, while closer objects usually have a high point

Table 7: Contribution of each component in our constructed GLENet-VR pipeline. “LU” denotes the label uncertainty.

KL loss	LU	var voting	UAQE	Easy	Moderate	Hard
				92.38	85.29	82.86
✓				92.45	85.25	82.99
✓		✓		92.48	85.37	83.05
✓	✓			93.20	85.76	83.29
✓	✓	✓		93.24	85.91	83.41
✓	✓	✓	✓	**93.49**	**86.10**	**83.56**

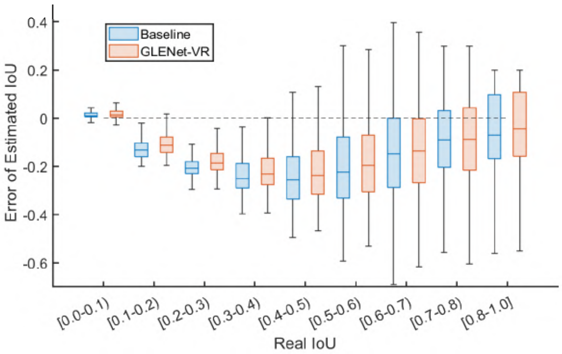


Fig. 10: Boxplots are used to display the estimated IoU error across various intervals of true IoU values. The x-axis represents the real IoU between proposals and their corresponding GT boxes, while the y-axis represents the distribution of estimation error, which is the difference between the estimated IoU score and the real IoU. The boxplot provides information about the distribution of error through five summary statistics: the minimum value, the maximum value, the median, the first quartile (Q1), and the third quartile (Q3).

count and low label uncertainty. For this reason, we performed experiments by adding the absolute coordinates of the point cloud as an extra feature in the input of GLENet. However, as shown in Table 8, the inclusion of extra absolute coordinates did not yield any significant improvement in the L_{NLL} metric or the performance of downstream detectors. We reason these observations from two aspects. First, the additional absolute coordinates may differentiate objects that are located in different positions but have similar appearances. As a result, there may be fewer samples with similar shapes but different bounding box labels, making it difficult for GLENet to capture the one-to-many relationship between incomplete point cloud objects and potential plausible bounding boxes. Second, the absolute distance and the point cloud density are generally correlated, i.e., an object with a larger absolute distance generally has a sparser point cloud representation, and such correlation could be perceived by the network. In other words, the absolute distance information is somewhat redundant to the network.

Table 8: Effect of point cloud input with and without absolute coordinates in GLENet. “NC” denotes normalized coordinates of the partial point cloud, and “AC” denotes absolute coordinates. We report the L_{NLL} for evaluation of GLENet and the 3D average precisions of 40 sampling recall points for evaluation of downstream detectors.

NC	AC	$L_{NLL}\downarrow$	Easy	Mod.	Hard	Avg
✓		**91.50**	**93.49**	**86.10**	**83.56**	**87.72**
✓	✓	147.33	93.21	85.66	83.35	87.41

Table 9: Ablation study on occlusion augmentation techniques and context encoder in GLENet, in which we report the L_{NLL} for evaluation of GLENet and the 3D average precisions of 40 sampling recall points for evaluation of downstream detectors.

Setting	$L_{NLL}\downarrow$	Easy	Mod.	Hard	Avg.
Baseline	**91.50**	**93.49**	**86.10**	**83.56**	**87.72**
w/o Occlusion Augmentation	230.10	92.96	85.52	83.07	87.18
w/o Context Encoder	434.93	92.65	85.31	82.59	86.85

Influence of Data Augmentation. To generate similar point cloud shapes with diverse ground-truth bounding boxes during training of GLENet, we proposed an occlusion data augmentation strategy and generated more incomplete point clouds while keeping the bounding boxes unchanged (see Fig. 8). As listed in Table 9, it can be seen that the occlusion data augmentation effectively enhances the performance of GLENet and the downstream detection task.

Necessity of the Context Encoder. In addition to learning the distribution of latent variables, the prior and recognition networks are also capable of extracting features from point clouds. To verify the necessity of the context encoder that is responsible for encoding contextual information from the input data in GLENet, we conducted an ablation experiment. As shown in Table 9, after removing the context encoder, we observed a significant deterioration in both the L_{NLL} metric and the average precision (AP) of the downstream detector. These results clearly demonstrate the necessity of the context encoder to extract geometric features from point clouds and allow the recognition and prior networks to focus on capturing the underlying structure of the input data in a low-dimensional space. Without the context encoder, the recognition and prior networks would need to learn both the geometric features and the contextual information from the input data, which would lead to poorer performance.

Dimension of the Latent Variable. Table. 10 shows the performance of adopting latent variables with various dimensions for GLENet. We can observe that the accuracy increase gradually, with the dimensions of latent variables from 2 to 8, and the setting of 32-dimensional latent variables achieve

Table 10: Ablation study of the dimensions of latent variables in GLENet.

Dimensions	$L_{NLL}\downarrow$	Easy	Mod.	Hard	Avg.
2	856.48	92.05	84.69	82.22	86.32
4	605.11	92.25	85.11	82.24	86.53
8	91.50	**93.49**	**86.15**	83.56	**87.73**
32	**86.16**	93.28	85.94	**83.60**	87.60
64	110.49	93.11	85.51	83.27	87.30
128	105.93	92.74	85.82	83.10	87.22

Table 11: Ablation study of the sampling times to calculate label uncertainty in GLENet.

Times	$L_{NLL}\downarrow$	Easy	Mod.	Hard	Avg.
4	608.82	92.54	85.11	81.21	86.29
8	240.08	92.96	85.52	82.80	87.09
16	148.21	92.99	85.66	83.35	87.33
30	91.5	93.49	86.10	**83.56**	**87.72**
64	86.76	93.37	**86.16**	83.42	87.65
128	**77.06**	**93.53**	85.92	83.47	87.64

similar performance. The results demonstrate a too-small dimension of the latent variables makes the GLENet unable to fully represent the underlying structure of the input data. And setting the dimension of latent variables to larger values like 64 or 128 can lead to over-fitting and slight decreases in performance. When the dimension of the latent variables is too large, the model can easily memorize the noise and details in the training data, which is not helpful for generating new and useful samples. Besides, though the setting of 32-dim latent variables leads to the lowest L_{NLL}, the performance of downstream detectors is best using label uncertainty with 8-dim latent variables. Therefore, though the L_{NLL} metric can reflect the quality of generating of GLENet to some extent, it is not guaranteed to be strongly correlated with the performance of downstream detectors.

Effects of the Sampling Times. In Table 11, we investigate the effects of the sampling times to calculate label uncertainty. We can observe that larger sampling times generally achieve lower L_{NLL} and better performance of downstream detectors, and similar performance is observed when using more than 30 sampling times. Statistically speaking, the variance obtained after a certain number of sampling times will tend to stabilize. Hence, to balance the computation cost and performance, we empirically choose to calculate the label uncertainty with predicted multiple bounding boxes by sampling the latent variables 30 times.

5.4.4 *Conditional Analysis*

To figure out in what cases our method improves the base detector most, we evaluated GLENet-VR on different occlusion levels and distance ranges. As shown in Table 12, compared with the baseline, our method mainly improves on the heavily occluded and distant samples, which suffer from more serious boundary ambiguities of ground-truth bounding boxes.

Table 12: Comparison on different occlusion levels and distance ranges[a], evaluated by the 3D Average Precision (AP) calculated with 40 sampling recall positions on the KITTI val set.

Methods		Voxel R-CNN (Deng et al., 2021)	GLENet-VR (Ours)	Improvement
Occlusion[b]	0	92.35	93.51	+1.16
	1	76.91	78.64	+1.73
	2	54.32	56.93	+2.61
Distance	0-20m	96.42	96.69	+0.27
	20-40m	83.82	86.87	+3.05
	40m-Inf	38.86	39.82	+0.96

[a] The results include separate APs for objects belonging to different occlusion levels and APs for the moderate vehicle class in different distance ranges.

[b] Definition of occlusion levels: levels 0, 1 and 2 correspond to fully visible samples, partly occluded samples, and samples difficult to see respectively.

Table 13: Inference time comparison for different baselines on the KITTI dataset.

Method	FPS (Hz)
SECOND Yan et al. (2018b)	23.36
GLENet-S (Ours)	22.80
CIA-SSD Zheng et al. (2021a)	27.18
GLENet-C (Ours)	28.76
Voxel R-CNN Deng et al. (2021)	21.08
GLENet-VR (Ours)	20.82

5.4.5 *Inference Efficiency*

We evaluated the inference speed of different baselines with a batch size of 1 on a desktop with Intel CPU E5-2560 @ 2.10 GHz and NVIDIA GeForce RTX 2080Ti GPU. As shown in Table 13, our approach does not significantly increase the computational overhead. Particularly, GLENet-VR only takes 0.6 more ms than the base Voxel R-CNN, since the number of candidates for the input of variance voting is relatively small in two-stage detectors.

5.5 Comparison of Visual Results

Fig. 11 visualizes the detection results of our GLENet-VR and the baseline Voxel R-CNN on the KITTI val set, where it can be seen that our GLENet-VR obtains better detection results with fewer false-positive bounding boxes and fewer missed heavily occluded and distant objects than Voxel R-CNN. We also compared the detection results of SECOND and GLENet-S on the Waymo validation set in Fig. 12, where it can be

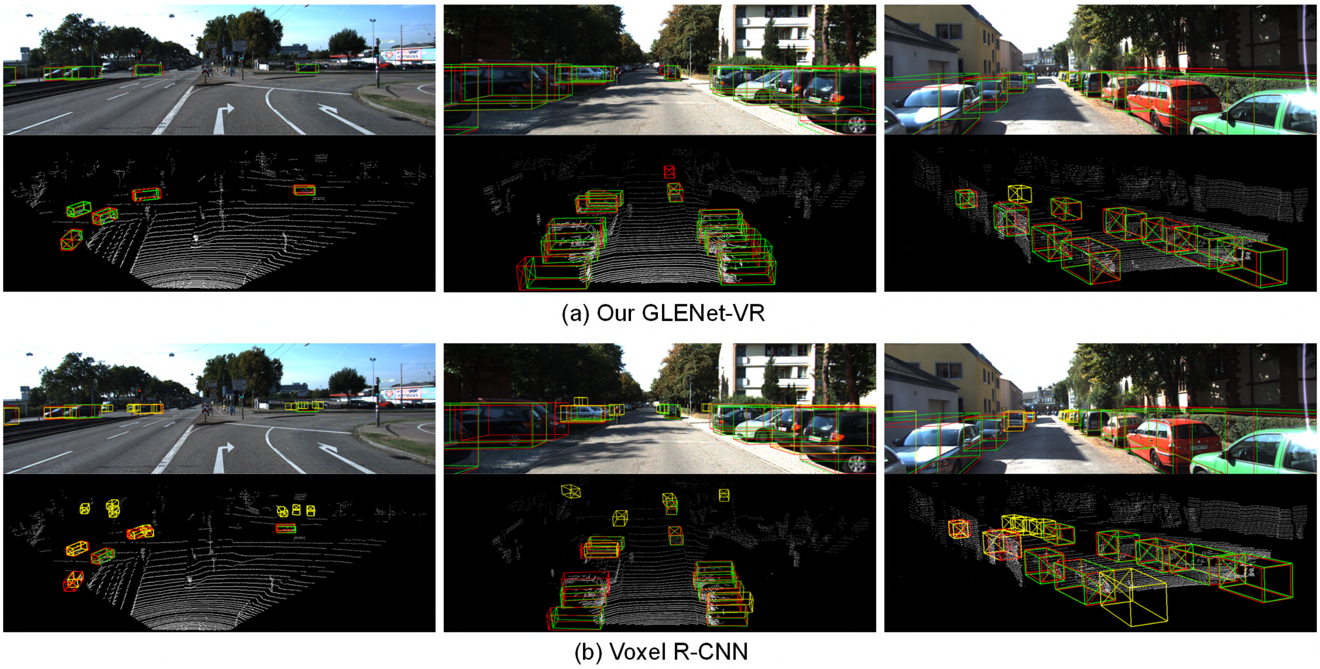


Fig. 11: Visual comparison of the results by GLENet-VR and Voxel R-CNN on the KITTI dataset. The ground-truth, true positive and false positive bounding boxes are visualized in red, green and yellow, respectively, on both the point cloud and image. Best viewed in color.

seen that compared with SECOND (Yan et al., 2018b), our GLENet-S has fewer false predictions and achieves more accurate localization.

6 Discussion

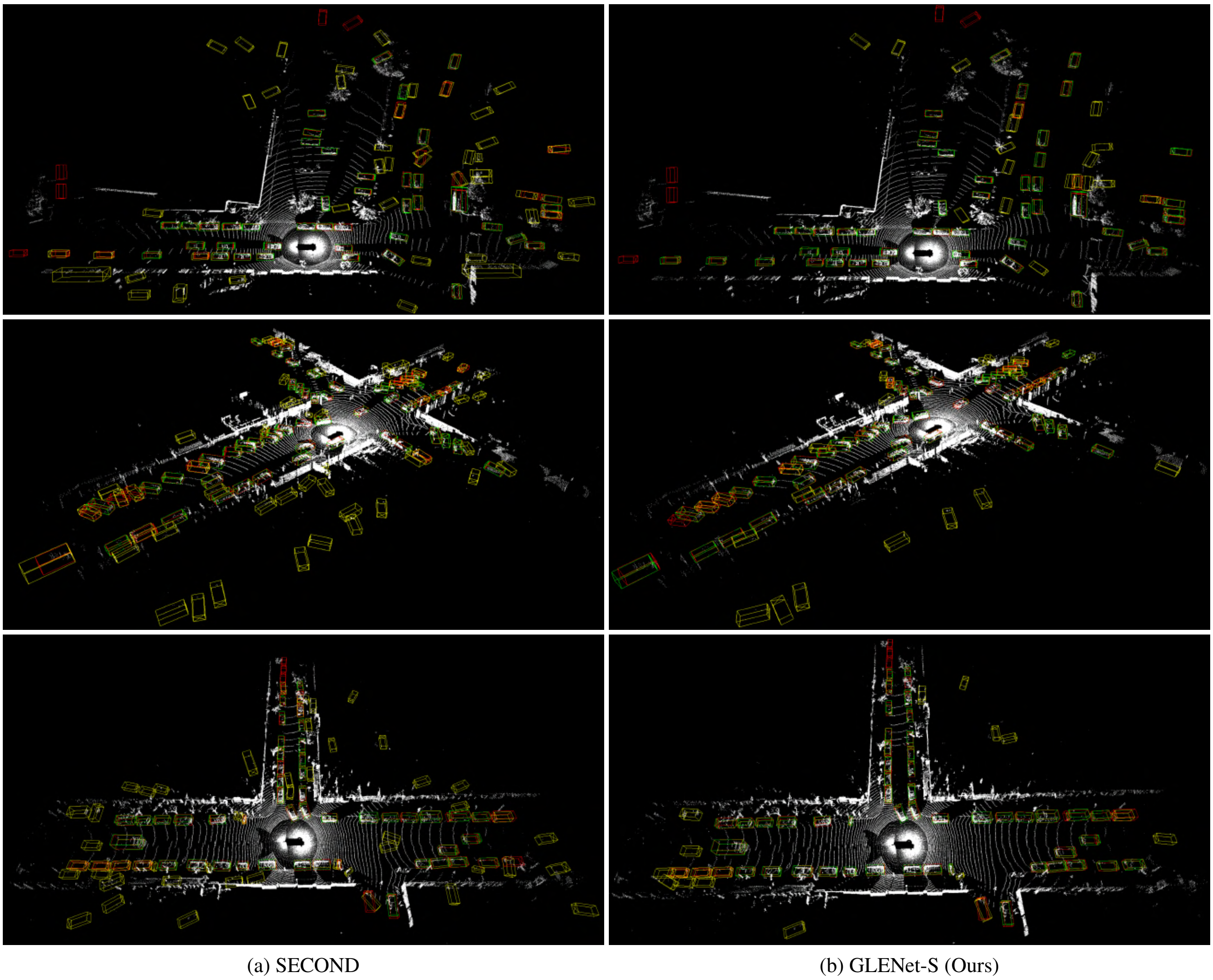
In this section, we further list a few potential technical limitations of the current learning framework and promising directions for extensions.

(1) *Complexity and Computational Cost*. Despite GLENet providing reliable label uncertainty as supervision signals for downstream probabilistic detectors, estimating the label uncertainty itself brings additional computational costs and makes the overall training process more complex. Particularly, considering the risk of over-fitting, we followed k-fold cross-sampling to train GLENet on 9 subsets and then made predictions on the remaining subset at each time.

(2) *Incomplete input information*. In GLENet, we only take the partial point cloud of individual objects as input, so only the learned geometric information is used to estimate potential bounding boxes. However, the context cues like free space and location of surrounding objects are neglected, which are also meaningful to determining the bounding boxes. Therefore, the estimated label uncertainty may deviate from the true distribution. But it is not feasible to take all points in the scene as input, as the key point of GLENet lies in learning the latent distribution of bounding boxes from samples with similar point cloud shapes and involving the whole point cloud in the scene distinguishes those objects with similar shapes. Incorporating such information without compromising the core benefits of GLENet remains a challenge.

(3) *Robustness to Annotation Errors*. While GLENet aims to address the inherent ambiguity in ground-truth annotations, it may not be entirely immune to the effects of significant annotation errors. If the training data contains substantial annotation errors, the model may inadvertently learn and propagate these errors, leading to an inaccurate estimation of label uncertainty. For example, if an object with a high-quality point cloud is annotated with a wrong box and further leads to inconsistent predictions and larger label uncertainty, those objects with similar shapes will suffer from unreasonable label uncertainty supervision signals. The robustness and reliability of the proposed method under such scenarios could be a limitation.

(4) *Limited Evaluation Metrics and Scenarios*. Evaluating the quality and diversity of generated data in generative tasks like GLENet is challenging. Although the proposed L_{NLL} assesses the closeness between the prediction of GLENet and ground-truth annotation bounding boxes, evaluating the quality and diversity of generated data remains an ongoing research problem. On the other hand, while your method demonstrates performance gains on benchmark datasets such as KITTI and Waymo, it is important to



(a) SECOND

(b) GLENet-S (Ours)

Fig. 12: Visual comparison of the results by SECOND and GLENet-S on the Waymo val set. The ground-truth, true positive and false positive bounding boxes are visualized in red, green and yellow, respectively. Best viewed in color and zoom in for more details. Additional NMS with a higher IoU threshold is conducted to eliminate overlapped bounding boxes for better visualization.

consider the generalizability of your approach across various environmental conditions, object classes, and sensor modalities. The ability of GLENet to generalize to a broader range of datasets and scenarios could be a limitation.

(5) *Possible Extensions*. The idea of estimating the label uncertainty by capturing the one-to-many relationship between observed input and multiple plausible labels with latent variables could be extended to other subjective tasks in computer vision where labels are not deterministic. One promising task is 3D object tracking, where different opinions of annotators on the boundaries of objects lead to non-deterministic labels. Another example is image quality assessment, where the goal is to evaluate the quality of an image, often in the context of compression or transmission. The quality of an image is subjective and can vary depending on the perception and expectations of the viewer.

7 Conclusion

We presented a general and unified deep learning-based paradigm for modeling 3D object-level label uncertainty. Technically, we proposed GLENet, adapted from the learning framework of CVAE, to capture one-to-many relationships between incomplete point cloud objects and potentially plausible bounding boxes. As a plug-and-play component, GLENet can generate reliable label uncertainty statistics that can be conveniently integrated into various 3D detection pipelines to build powerful probabilistic detectors. We verified the effectiveness and universality of our method by incorporating the proposed GLENet into several existing deep 3D object detectors, which demonstrated consistent improvement and produced state-of-the-art performance on both KITTI and Waymo datasets.

Data Availability Statements

The Waymo Open Dataset (Sun et al., 2020) and KITTI (Geiger et al., 2012) used in this manuscript are deposited in publicly available repositories respectively: https://waymo.com/open/data/perception and http://www.cvlibs.net/datasets/kitti.

References

Bowman S, Vilnis L, Vinyals O, Dai A, Jozefowicz R, Bengio S (2016) Generating sentences from a continuous space. In: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, pp 10–21

Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S (2020) End-to-end object detection with transformers. In: European conference on computer vision, pp 213–229

Chen X, Ma H, Wan J, Li B, Xia T (2017) Multi-view 3d object detection network for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1907–1915

Choi J, Chun D, Kim H, Lee HJ (2019) Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving. In: Proceedings of the IEEE International Conference on Computer Vision, vol 2019-October, pp 502–511

Delany SJ, Segata N, Mac Namee B (2012) Profiling instances in noise reduction. Knowledge-Based Systems 31:28–40

Deng J, Shi S, Li P, Zhou W, Zhang Y, Li H (2021) Voxel r-cnn: Towards high performance voxel-based 3d object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp 1201–1209

Feng D, Rosenbaum L, Dietmayer K (2018) Towards safe autonomous driving: Capture uncertainty in the deep neural network for lidar 3d vehicle detection. In: IEEE Conference on Intelligent Transportation Systems, pp 3266–3273

Feng D, Rosenbaum L, Timm F, Dietmayer K (2019) Leveraging heteroscedastic aleatoric uncertainties for robust real-time lidar 3d object detection. In: IEEE Intelligent Vehicles Symposium, pp 1280–1287

Garcia LP, Sáez JA, Luengo J, Lorena AC, de Carvalho AC, Herrera F (2015) Using the one-vs-one decomposition to improve the performance of class noise filters via an aggregation strategy in multi-class classification problems. Knowledge-Based Systems 90:153–164

Geiger A, Lenz P, Urtasun R (2012) Are we ready for autonomous driving? the kitti vision benchmark suite. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3354–3361

Goyal A, Sordoni A, Côté MA, Ke N, Bengio Y (2017) Z-forcing: Training stochastic recurrent networks. In: Advances in Neural Information Processing Systems, pp 6714–6724

Harakeh A, Smart M, Waslander SL (2020) Bayesod: A bayesian approach for uncertainty estimation in deep object detectors. In: IEEE International Conference on Robotics and Automation, IEEE, pp 87–93

He C, Zeng H, Huang J, Hua XS, Zhang L (2020) Structure aware single-stage 3d object detection from point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 11870–11879

He Y, Zhu C, Wang J, Savvides M, Zhang X (2019) Bounding box regression with uncertainty for accurate object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2888–2897

Huang T, Liu Z, Chen X, Bai X (2020) Epnet: Enhancing point features with image semantics for 3d object detection. In: European Conference on Computer Vision, Springer, pp 35–52

Kingma D, Ba J (2015) Adam: A method for stochastic optimization. In: International Conference on Learning Representations, pp 1–15

Kingma DP, Welling M (2014) Auto-encoding variational bayes. In: International Conference on Learning Representations, pp 1–14

Lang A, Vora S, Caesar H, Zhou L, Yang J, Beijbom O (2019) Pointpillars: Fast encoders for object detection from point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 12689–12697

Li B, Sun Z, Guo Y (2019) Supervae: Superpixelwise variational autoencoder for salient object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 33, pp 8569–8576

Li J, Song Y, Zhang H, Chen D, Shi S, Zhao D, Yan R (2018) Generating classical chinese poems via conditional variational autoencoder and adversarial training. In: Proceedings of the 2018 conference on empirical methods in natural language processing, pp 3890–3900

Li X, Wang W, Hu X, Li J, Tang J, Yang J (2021) Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 11632–11641

Liang M, Yang B, Chen Y, Hu R, Urtasun R (2019) Multi-task multi-sensor fusion for 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7345–7353

Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) Ssd: Single shot multibox detector. In: European conference on computer vision, Springer, pp 21–37

Luengo J, Shim SO, Alshomrani S, Altalhi A, Herrera F (2018) Cnc-nos: Class noise cleaning by ensemble filtering and noise scoring. Knowledge-Based Systems 140:27–49

Mao J, Niu M, Bai H, Liang X, Xu H, Xu C (2021a) Pyramid r-cnn: Towards better performance and adaptability for 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 2723–2732

Mao J, Xue Y, Niu M, Bai H, Feng J, Liang X, Xu H, Xu C (2021b) Voxel transformer for 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 3164–3173

Meyer G, Thakurdesai N (2020) Learning an uncertainty-aware object detector for autonomous driving. In: IEEE International Conference on Intelligent Robots and Systems, pp 10521–10527

Meyer GP, Laddha A, Kee E, Vallespi-Gonzalez C, Wellington CK (2019) Lasernet: An efficient probabilistic 3d object detector for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 12677–12686

Mousavian A, Eppner C, Fox D (2019) 6-dof graspnet: Variational grasp generation for object manipulation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 2901–2910

Najibi M, Lai G, Kundu A, Lu Z, Rathod V, Funkhouser T, Pantofaru C, Ross D, Davis L, Fathi A (2020) Dops: Learning to detect 3d objects and predict their 3d shapes. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 11910–11919

Nash C, Williams C (2017) The shape variational autoencoder: A deep generative model of part-segmented 3d objects. Computer Graphics Forum 36(5):1–12

Northcutt C, Jiang L, Chuang I (2021) Confident learning: Estimating uncertainty in dataset labels. Journal of Artificial Intelligence Research 70:1373–1411

Painchaud N, Skandarani Y, Judge T, Bernard O, Lalande A, Jodoin PM (2020) Cardiac segmentation with strong anatomical guarantees. IEEE transactions on medical imaging 39(11):3703–3713

Pang S, Morris D, Radha H (2020) Clocs: Camera-lidar object candidates fusion for 3d object detection. In: IEEE International Conference on Intelligent Robots and Systems, IEEE, pp 10386–10393

Qi CR, Su H, Mo K, Guibas LJ (2017) Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 652–660

Qi CR, Liu W, Wu C, Su H, Guibas LJ (2018) Frustum pointnets for 3d object detection from rgb-d data. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 918–927

Sharma S, Varigonda PT, Bindal P, Sharma A, Jain A (2019) Monocular 3d human pose estimation by generation and ordinal ranking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 2325–2334

Sheng H, Cai S, Liu Y, Deng B, Huang J, Hua XS, Zhao MJ (2021) Improving 3d object detection with channel-wise transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 2743–2752

Shi S, Wang X, Li H (2019) Pointrcnn: 3d object proposal generation and detection from point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 770–779

Shi S, Guo C, Jiang L, Wang Z, Shi J, Wang X, Li H (2020a) Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10529–10538

Shi S, Wang Z, Shi J, Wang X, Li H (2020b) From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. IEEE Transactions on Pattern Analysis and Machine Intelligence 43(8):2647–2664

Shi W, Rajkumar R (2020a) Point-gnn: Graph neural network for 3d object detection in a point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1708–1716

Shi W, Rajkumar R (2020b) Point-gnn: Graph neural network for 3d object detection in a point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1711–1719

Smith LN (2017) Cyclical learning rates for training neural networks. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, IEEE, pp 464–472

Sohn K, Lee H, Yan X (2015) Learning structured output representation using deep conditional generative models. In: Advances in Neural Information Processing Systems, pp 3483–3491

Sun P, Kretzschmar H, Dotiwalla X, Chouard A, Patnaik V, Tsui P, Guo J, Zhou Y, Chai Y, Caine B, Vasudevan V, Han W, Ngiam J, Zhao H, Timofeev A, Ettinger S, Krivokon M, Gao A, Joshi A, Zhang Y, Shlens J, Chen Z, Anguelov D (2020) Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2443–2451

Tan M, Pang R, Le QV (2020) Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10781–10790

Varamesh A, Tuytelaars T (2020) Mixture dense regression for object detection and human pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 13086–13095

Vora S, Lang AH, Helou B, Beijbom O (2020) Pointpainting: Sequential fusion for 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4604–4612

Wang T, Wan X (2019) T-cvae: Transformer-based conditioned variational autoencoder for story completion. In: International Joint Conference on Artificial Intelligence, pp 5233–5239

Wang Z, Feng D, Zhou Y, Rosenbaum L, Timm F, Dietmayer K, Tomizuka M, Zhan W (2020) Inferring spatial uncertainty in object detection. In: IEEE International Conference on Intelligent Robots and Systems, IEEE, pp 5792–5799

Xu Q, Zhou Y, Wang W, Qi CR, Anguelov D (2021) Spg: Unsupervised domain adaptation for 3d object detection via semantic point generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 15446–15456

Yan X, Yang J, Sohn K, Lee H (2016) Attribute2image: Conditional image generation from visual attributes. In: European conference on computer vision, Springer, pp 776–791

Yan X, Rastogi A, Villegas R, Sunkavalli K, Shechtman E, Hadap S, Yumer E, Lee H (2018a) Mt-vae: Learning motion transformations to generate multimodal human dynamics. In: European conference on computer vision, pp 265–281

Yan X, Gao J, Li J, Zhang R, Li Z, Huang R, Cui S (2021) Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 35, pp 3101–3109

Yan Y, Mao Y, Li B (2018b) Second: Sparsely embedded convolutional detection. Sensors 18(10):3337

Yang B, Luo W, Urtasun R (2018) Pixor: Real-time 3d object detection from point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7652–7660

Yang Z, Sun Y, Liu S, Shen X, Jia J (2019) Std: Sparse-to-dense 3d object detector for point cloud. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 1951–1960

Yang Z, Sun Y, Liu S, Jia J (2020) 3dssd: Point-based 3d single stage object detector. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 11040–11048

Yi L, Zhao W, Wang H, Sung M, Guibas L (2019) Gspn: Generative shape proposal network for 3d instance segmentation in point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3942–3951

Yin T, Zhou X, Krahenbuhl P (2021) Center-based 3d object detection and tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 11784–11793

Yoo JH, Kim Y, Kim J, Choi JW (2020) 3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection. In: European Conference on Computer Vision, Springer, pp 720–736

Zhang B, Xiong D, Su J, Duan H, Zhang M (2016) Variational neural machine translation. In: Proceedings of the 2016 conference on empirical methods in natural language processing, Association for Computational Linguistics, Austin, Texas, pp 521–530

Zhang C, Bengio S, Hardt M, Recht B, Vinyals O (2021) Understanding deep learning (still) requires rethinking generalization. Communications of the ACM 64(3):107–115

Zhang J, Fan DP, Dai Y, Anwar S, Saleh FS, Zhang T, Barnes N (2020) Uc-net: Uncertainty inspired rgb-d saliency detection via conditional variational autoencoders. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 8582–8591

Zhao T, Zhao R, Eskenazi M (2017) Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In: ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, pp 654–664

Zheng W, Tang W, Chen S, Jiang L, Fu CW (2021a) Cia-ssd: Confident iou-aware single-stage object detector from point cloud. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp 3555–3562

Zheng W, Tang W, Jiang L, Fu CW (2021b) Se-ssd: Self-ensembling single-stage object detector from point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 14494–14503

Zhou Y, Tuzel O (2018) Voxelnet: End-to-end learning for point cloud based 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4490–4499

Zhou Y, Sun P, Zhang Y, Anguelov D, Gao J, Ouyang T, Guo J, Ngiam J, Vasudevan V (2020) End-to-end multi-view fusion for 3d object detection in lidar point clouds. In: Conference on Robot Learning, PMLR, pp 923–932