# Probabilistic and Geometric Depth: Detecting Objects in Perspective

## *Supplementary Materials*

**Tai Wang**[1,2] **Xinge Zhu**[1] **Jiangmiao Pang**[1,2*] **Dahua Lin**[1,2,3]
[1]CUHK-SenseTime Joint Lab, The Chinese University of Hong Kong
[2]Shanghai AI Laboratory [3]Centre of Perceptual and Interactive Intelligence
{wt019, zx018, dhlin}@ie.cuhk.edu.hk, pangjiangmiao@gmail.com

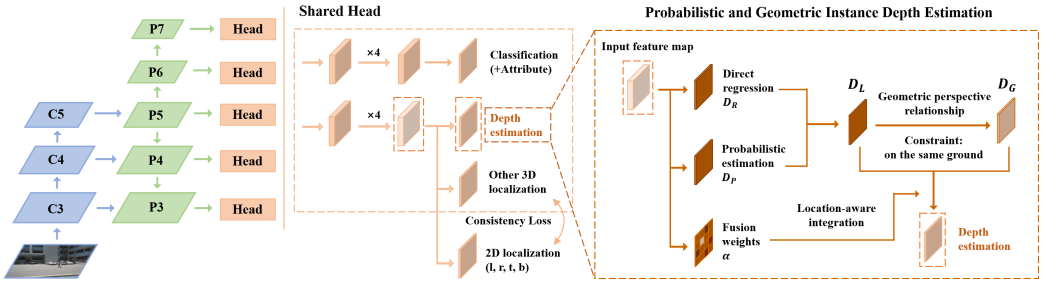


Figure 1: An overview of our framework (Figure 3 in the main paper).

## 1 Implementation Details

This section first presents the adopted local geometric constraints between 2D and projected 3D bounding boxes in the enhanced baseline. Subsequently, we will elaborate on the details of training loss and inference procedure.

### 1.1 Local Geometric Constraints

Our baseline FCOS3D [1] only stiffly adjusts the output of networks to fit the requirements of 3D detection. There is no relationship or constraints between these predicted attributes, making this network hard to train, especially when the data is limited. Considering our detector can achieve 90% accuracy on 2D vehicle detection, we add 2D localization into our targets and use it to regularize 3D outputs. Actually, this closed-loop and self-supervised approach is also consistent with what humans do in the annotation procedure [2]. In practice, as shown in the Fig. 1, we add a consistency loss (GIoU loss) between our estimated 2D boxes and the exterior 2D boxes of 3D predictions to enhance our baseline, which is particularly important on the small KITTI dataset. Note that due to the difficulty of regressing accurate depth, we use the ground truth depth for deriving the 3D bounding boxes when computing the consistency loss.

Here we provide an example to show the intuition behind this design. Typically when the data is limited, it is hard for the network to direct regress different 3D targets (offset, depth, orientation, *etc.*) independently. For example, in Fig. 2, the orientation of nearby large objects predicted by our baseline can be very inaccurate (the top line in the figure) even though it can be easily rectified with simple verification. So we add the more reliable 2D localization into our targets to regularize our 3D predictions. It turns out that the simple local constraint could alleviate this problem in the learning procedure while does not introduce extra computational costs to inference. The improved results after adding this constraint can be seen in Fig. 2 (the bottom line).

### 1.2 Loss

**Overall Loss Design** We basically follow the loss design of FCOS3D except our proposed consistency loss and the adjustments for different datasets.

To have a brief review, firstly, we use the focal loss [3] as the object classification loss:

$$L_{cls} = -\alpha(1-p)^{\gamma} log p \tag{1}$$

*Corresponding author

5th Conference on Robot Learning (CoRL 2021), London, UK.

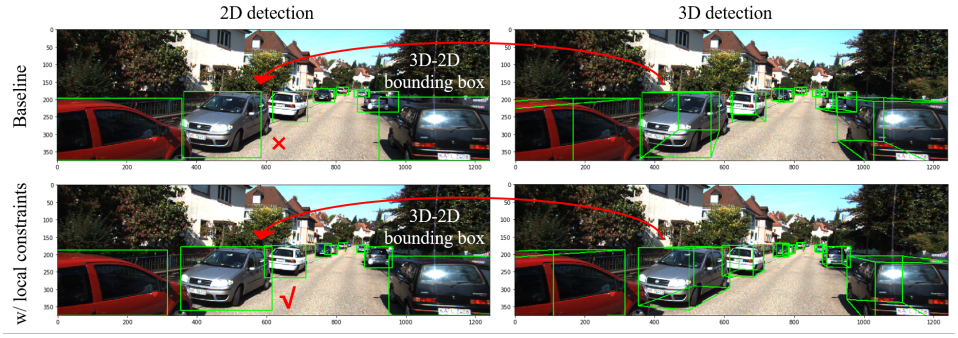


Figure 2: The top line shows that it is easy to validate the accuracy of 3D predictions according to its exterior 2D bounding box. So we add the 2D localization into our targets and use the relatively reliable 2D boxes to regularize 3D predictions. This results in significant improvement as shown by the bottom line.

where $p$ is the class probability of a predicted box, and we follow the common settings, $\alpha = 0.25$ and $\gamma = 2$. For attribute classification on nuScenes, we use a simple softmax classification loss, denoted as $L_{attr}$.

For regression branch, we use the smooth L1 loss for each regression target except centerness:

$$L_{loc} = \sum_{b \in (\Delta x, \Delta y, d, w, l, h, \theta, v_x, v_y)} SmoothL1(\Delta b) \tag{2}$$

The weights of $\Delta x, \Delta y, d, w, l, h, \theta$ error are 1 and the weights of $v_x, v_y$ on nuScenes are 0.05. We use the softmax classification loss and binary cross entropy (BCE) loss for direction classification and centerness regression, denoted as $L_{dir}$ and $L_{ct}$ respectively. For local geometric constraints, denote our predicted 2D boxes as $\boldsymbol{B_{2D}}$, the minimum exterior 2D boxes of projected 3D boxes as $\boldsymbol{B_{proj}}$, then the consistency loss is:

$$L_{geo} = GIoU(\boldsymbol{B_{2D}}, \boldsymbol{B_{proj}}) \tag{3}$$

Finally, the total loss is:

$$L = \frac{1}{N_{pos}}(\beta_{cls} L_{cls} + \beta_{attr} L_{attr} + \beta_{loc} L_{loc} + \beta_{dir} L_{dir} + \beta_{ct} L_{ct} + \beta_{geo} L_{geo}) \tag{4}$$

$N_{pos}$ is the number of positive predictions and $\beta_{cls} = \beta_{attr} = \beta_{loc} = \beta_{dir} = \beta_{ct} = \beta_{geo} = 1$. Note that the attribute loss $L_{attr}$ and velocity loss in the $L_{loc}$ are only required in the nuScenes experiments.

**Specific Loss Designs for KITTI experiments** Because the KITTI dataset has relatively limited samples and much more strict metrics, we adopt two specific loss designs for training the networks. First, we add an auxiliary key-points loss to enhance the local geometric consistency further. Denote the 2D offsets of eight key-points (eight corners of a 3D bounding box) relative to a foreground point as $\boldsymbol{k} \in \mathbb{R}^{1\times 16}$, and then we take these offsets as 16 additional dimensions of $b$ in Eqn. 2 and set their weights to 0.2. To make the FPN-based learning stable, we normalize these offsets just as we normalize those offsets to four sides of a 2D box.

In addition, we use a much stronger uncertainty formulation for this multi-task learning problem as presented in [4]. Specifically, referring to its formulation of maximum likelihood and homoscedastic uncertainty, we formulate the depth loss as:

$$L_{depth} = \frac{L_1(\hat{D}, D)}{2\sigma^2} + log\sigma \tag{5}$$

Here $\hat{D}$ and $D$ are the targets and predictions of depth, $L_1$ represents the original smooth L1 loss with $\delta = 3.0$ and $\sigma$ is the variable for uncertainty. In practice, to make the learning easier, we train the network to predict the log variance $s = log\sigma^2$ only for depth estimation, which is more numerically stable than directly predicting the variance. Correspondingly, $exp(-s)$ serves as the weight of depth loss. In this way, the depth loss will be adaptively weighted relative to other regression losses. Additionally, the uncertainty $exp(-s)$ can also be used as another confidence score to be multiplied when inference, such that predictions with more accurate depths will have particularly higher scores.

Note that this strong uncertainty indicator can only bring a significant gain on KITTI experiments while seriously hurting the general performance as evaluated on the nuScenes dataset.

**Alternative Depth Loss Designs** Considering we have several intermediate depth predictions, such as $D_R$, $D_P$ and $D_L$ in Fig. 1, a natural idea is to add intermediate supervisions for these predictions to guarantee that each branch can learn meaningful information. So we further defined several depth L1 losses for these predictions and tried to replace the original depth loss in the $L_{loc}$ with their weighted summation. It turns out that although this approach can make the training procedure more stable, it does not bring any performance gain. We also find that the framework never overfits to only relying on one kind of estimation even with only supervision for the final prediction, as to be shown in Sec. 3.2. It indicates that these predictions and components indeed work together from complementary aspects.

### 1.3 Inference

The inference procedure is to forward the input image through the framework and obtain bounding boxes with their class scores, attribute scores (if necessary) and centerness predictions. We multiply the class score, the predicted centerness and the depth confidence score as the overall confidence for each prediction and conduct rotated Non-Maximum Suppression (NMS) in the bird view as most 3D detectors to get the final results.

## 2 Explanation of Oracle Analyses

In this section, we will explain more about our empirical analysis, from the specific settings to more details in the results.

### 2.1 Reason for Replacing Dense Predictions

First, we would like to emphasize one detail in our analysis, *i.e.*, we replace the dense predictions from the direct output of detection head with oracles to purely observe the problems of our networks. In comparison, other alternatives exist, such as replacing the decoded dense output or predictions after post-processing, which can not reveal some entangling problem lying in the formulation. One example to show the difference between these two implementations is that we replace the offset with corresponding ground truth while the latter approach replaces the decoded $X, Y$ in the 3D space with targets.

### 2.2 Comparison of Different Metrics

As mentioned in the main paper, KITTI and nuScenes adopt different evaluation metrics. The former is relatively strict and the latter is more comprehensive. Specifically, for mAP of these two datasets, we regard predictions with 3D IoU larger than a threshold (0.7 or 0.5) as positive samples on KITTI while define the match by 2D center distance $d_{2D}$ in the bird eye view on nuScenes. The latter is a simpler criterion as it decouples the detection from object size and orientation. Therefore, we only plot points with category/location related oracles (classification, depth and offset) in the mAP analysis on nuScenes (Fig. 3). In addition, to be more specific, mAP is computed over several different matching thresholds, $\mathbb{D} = \{0.5, 1, 2, 4\}$ meters, and all categories $\mathbb{C}$ on nuScenes:

$$mAP = \frac{1}{|\mathbb{C}||\mathbb{D}|} \sum_{c \in \mathbb{C}} \sum_{d_{2D} \in \mathbb{D}} AP_{c,d_{2D}} \tag{6}$$

Then we can see that it will also consider predictions with relatively inaccurate locations (like objects with the distance error larger than 2 meters but smaller than 4 meters). This difference is especially notable when discussing the improvements from depth score, which will be detailed in Sec. 3.2.

Finally we basically describe how the NuScenes Detection Score (NDS) is calculated. To begin with, we first define that predictions with center distance from the matching ground truth $d_{2D} \leq 2m$ will be considered as true positives (TP) and thus introduce 5 True Positive metrics, Average Translation Error (ATE), Average Scale Error (ASE), Average Orientation Error (AOE), Average Velocity Error (AVE) and Average Attribute Error (AAE). Given these metrics, we compute the mean TP metric (mTP) over all categories:

$$mTP = \frac{1}{|\mathbb{C}|} \sum_{c \in \mathbb{C}} TP_c \tag{7}$$

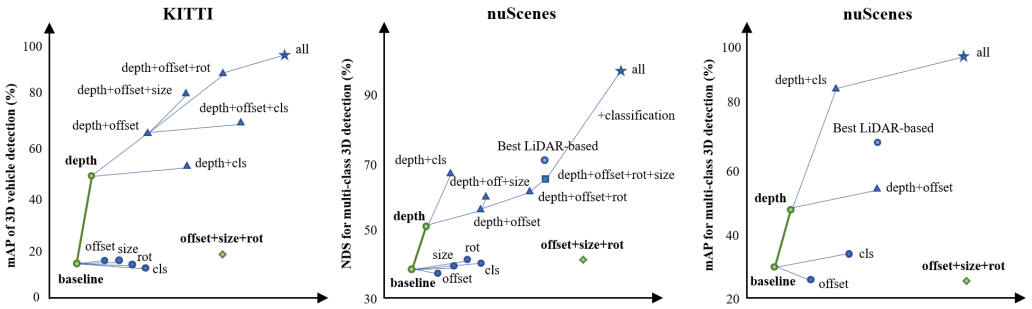


Figure 3: Oracle analyses with different datasets and metrics (Figure 2 in the main paper). From left to right: 3D IoU based mAP on KITTI, NuScenes Detection Score (NDS) and distance-based mAP on nuScenes. We replace our predictions with ground truth values step by step and observe the performance improvements. It can be seen that an accurate depth can bring significant performance improvement (green lines), and only with accurate depth can the improvements brought by other oracles be realized.

Then the NDS is calculated as follows:

$$NDS = \frac{1}{10}[5mAP + \sum_{mTP \in \mathbb{TP}} (1 - min(1, mTP))] \tag{8}$$

Therefore, NDS is a combination of several decoupled metrics and could reflect the performance of 3D detectors from another perspective. See more details about the intermediate computation in its original paper [5].

### 2.3 Detailed Explanations and Conclusions

Due to the space limitation in the main paper, we do not discuss much about the results shown in Fig. 3. Next, we will analyze it in detail and summarize a series of important conclusions.

**Basic Observations** As shown in Fig. 3, we replace the predicted attributes with their ground truth values step by step and observe the performance improvements. We can see that:

1. With only one oracle (circle dots), only depth can bring a considerable improvement (green lines). It shows that with current depth estimation, other predicted attributes do not drag down the performance, while with other predictions, the current accuracy of depth estimation is far not enough.

2. With accurate depth, other oracles (triangle dots in the figures) could bring the expected performance gains. While with current depth estimation, even all the other predictions are accurate (green rhombus dots), the results are always disappointing, even almost like the baseline.

3. Although KITTI and nuScenes are different in terms of category variety and metrics, the trend of these curves is the same. The difference is reflected in the importance of localization and classification oracles. Localization is more important on the KITTI, which has less category variety and more strict metrics. Classification is another important factor apart from depth on nuScenes, *e.g.*, our monocular predictions with location oracle is still not better than the best LiDAR-based methods. In contrast, with an accurate depth and classification map, the performance is almost ideal.

From these observations, we can conclude that the inaccurate depth blocks *all* the other sub-task predictions from improving the overall detection performance. Hence, as mentioned in the main paper, the current monocular 3D detection, especially 3D localization, can be actually reduced to the dominating instance depth estimation problem.

**Comparison with Best LiDAR-Based Methods** There is an interesting phenomenon not much related to depth estimation in the above analysis, *i.e.*, the comparison with best LiDAR-based methods on nuScenes. We can see that classification is particularly important on nuScenes, and our monocular predictions with location oracles are still not better than the state-of-the-art LiDAR-based methods. This result is a little dataset-specific. We conjecture it is because the classification for ten categories on nuScenes is relatively hard, or the annotation is mainly conducted in the point clouds, leading to missing objects in the images.


Figure 4: Inconsistent bottoms of different instances. Although all the objects in an image share similar heights for bottoms most of the time, corner cases still exist. Here we mark the heights of bottoms in the camera coordinates (down is the positive direction). This problem can be caused by the actual topography, *e.g.*, pedestrians are on the step. It can also be caused by annotation noises, especially for different categories and distant objects. This observation is the foundation of our proposed edge pruning/gating scheme in the depth propagation.

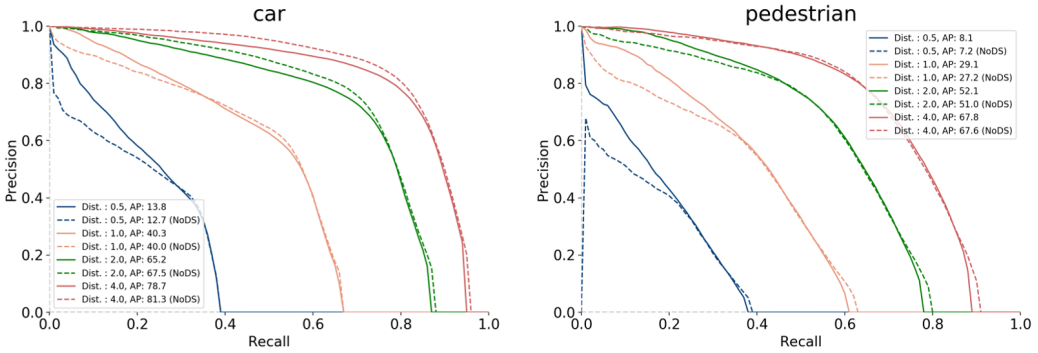


Figure 5: Comparison of PR curves for models with (solid line) and without (dotted line) depth score. The depth score encourages predictions with accurate depth while suppresses those with inaccurate depth, which results in higher precision under low recall and strict matching thresholds while lower precision under high recall. This problem is more notable for large objects like cars.

## 3 Supplementary Experimental Results

In this section, we will show more experimental results to help further understand our approach. First, we will provide toy examples to explain and validate our derived pairwise perspective relationship in the depth propagation. Subsequently, we make more detailed analyses in quantitative and qualitative ways to reveal the working mechanism and effect of our method.

### 3.1 Basic Validation of Depth Propagation

As shown in Fig. 4, we provide two samples with many objects in one image. We first have a brief review of the perspective relationship derived in the main paper. Given two objects 1 and 2, the relationship between their centers strictly satisfies:

$$d_2 = \frac{v_1}{v_2} d_1 + \frac{f}{v_2}(y_2 - y_1) \tag{9}$$

Considering two objects share the same ground (bottom height), we can get the approximate relationship as follows:

$$d_2 = \frac{v_1}{v_2} d_1 + \frac{f}{2v_2}(h_1^{3D} - h_2^{3D}) \tag{10}$$

where $d$ denotes the depth, $v$ denotes the distance between the projected 2D object center and the horizon line in the image, $y$ is the 3D height of object center and $h^{3D}$ is the height of the 3D bounding box. Taking the left sample in Fig. 4 as an example, the depths of the 8 cars are $\{5.23, 11.80, 16.50, 22.05, 23.64, 28.53, 29.07, 42.85\}$. With

Table 1: Average precision for each class on the nuScenes test benchmark. CV and TC are abbreviation of construction vehicle and traffic cone in the table.

| Methods | car | truck | bus | trailer | CV | ped | motor | bicycle | TC | barrier | mAP |
|---|---|---|---|---|---|---|---|---|---|---|---|
| LRM0 | 0.467 | 0.21 | 0.17 | 0.149 | 0.061 | 0.359 | 0.287 | 0.246 | 0.476 | 0.512 | 0.294 |
| MonoDIS [6] | 0.478 | 0.22 | 0.188 | 0.176 | 0.074 | 0.37 | 0.29 | 0.245 | 0.487 | 0.511 | 0.304 |
| CenterNet [7] (HGLS) | 0.536 | 0.27 | 0.248 | 0.251 | 0.086 | 0.375 | 0.291 | 0.207 | 0.583 | 0.533 | 0.338 |
| Noah CV Lab | 0.515 | 0.278 | 0.249 | 0.213 | 0.066 | 0.404 | 0.338 | 0.237 | 0.522 | 0.49 | 0.331 |
| PGD (Ours) | 0.561 | 0.299 | 0.285 | 0.266 | 0.134 | 0.441 | 0.397 | 0.314 | 0.605 | 0.561 | **0.386** |

our derived relationship, we can estimate them with only the first 2 accurate depths: $\{5.23, 11.74, 16.78, 22.92, 21.13, 26.59, 25.78, 36.51\}$. We can see that similar to the general case of depth estimation, our propagation mechanism also yields more notable errors for distant objects, which has been analyzed in the main paper (The effect of $\delta$ over $\Delta d$ will be enlarged as the $v_2$ decreases.)

Next, we can further observe the inconsistent bottoms problem shown in Fig. 4. We mark some representative instances in the figure. It can be seen that it is sometimes caused by the actual topography, like pedestrians and cars in the second sample. Nevertheless, the noise only exists between objects far away from each other most of the time. We conjecture this is related to the annotation pipeline, *e.g.*, we tend to make use of nearby annotations when the information for labeling the current instance is inadequate. Alternatively, sometimes it is just because the LiDAR only sweeps the top part of the distant objects such that the annotator can not determine its bottom accurately.

In conclusion, although the ground constraint holds most of the time, it is still important to design mechanisms to avoid these possible noises and incorporate the geometric depth adaptively, such as the edge pruning/gating scheme and location-aware integration in the main paper.

### 3.2 Quantitative Analysis

**Difference Between Datasets** Here we mainly show the observation in the ablation study to explain the different effects from the same component on these two datasets. We take the depth score as an example. First, Tab. 7 in the main paper has shown the especially important role of depth score on the KITTI. However, it does not contribute much to the improvements on nuScenes. Specifically, it only brings about 0.3% increase on NDS by reducing the mATE instead of boosting the mAP. To figure out the reason, we take a closer look at the performance from the Precision-Recall (PR) curve. As shown in Fig. 5, we can see that the depth score (solid line) significantly improves the precision under low recall and strict matching thresholds (like 0.5 and 1.0 meters, blue and yellow lines) while influences the performance under high recall and less strict cases (like 2.0 and 4.0 meters, green and red lines). This problem is especially notable for large objects. It reveals the effect of depth score from another perspective, *i.e.*, it can overly suppress those predictions with inaccurate depth, of which we should be tolerant under some circumstances, like distant and small objects. Therefore, designing a more suitable depth score with better interval division methods or other approaches can be a direction worthy of further exploration.

**Mean AP for Multi-Class Detection on nuScenes** To present the multi-class detection results more comprehensively, we provide the mean AP results (over all the matching thresholds) for each category on nuScenes in Tab. 1. We can see that our method shows the superiority especially on small (from pedestrian to barrier) and quite large objects (bus). Firstly, the better capability of handling objects with different scales should partly come from the leveraged well-developed backbone and FPN. Furthermore, our probabilistic and geometric depth also improves the accuracy of depth estimation, which is especially important for small objects.

**Contributions of Each Depth Estimation** To understand the role of each component for depth estimation more clearly, we make statistics about the fusion weights. Firstly, for local depth estimation, we find that the direct regression accounts for about 25.6% in the results, *i.e.*, $\sigma(\lambda)$ is about 0.256. It implies that the direct regression may be responsible for regressing the residual of the probabilistic estimation, which plays an auxiliary but important role according to the ablation study in the main paper (Tab. 7). On the other hand, for final integration, we make statistics for the location-aware weights $\sigma(\alpha)$ of predictions with matching ground truths before NMS on the validation set, and plot its distribution in Fig. 6 (higher value means more contribution from local estimation). We can see that although the preliminary local estimation plays a more important role in many cases, the propagated geometric depth does contribute a lot to the overall estimation. In addition, we also plot the scatter diagram of these weights with respect to the estimated depth and different categories (Fig. 7

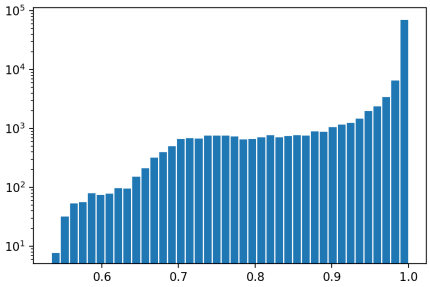

Figure 6: Distribution of weights for the final integration in our PGD module.

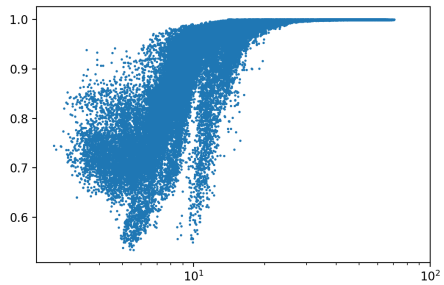

Figure 7: Scatter plot of location-aware weights with respect to depths.

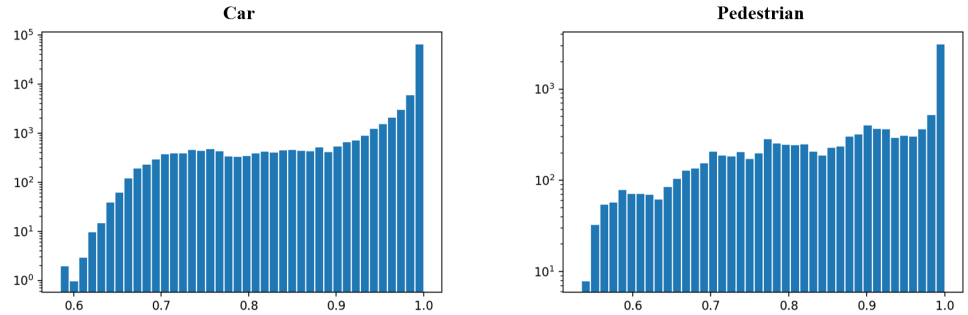


Figure 8: Location-aware weights of predictions from different categories.

and 8). We can see that the geometric depth contributes more to the estimation of very nearby (can be truncated in the image) and small objects like pedestrians, which is consistent with our common sense that these two cases are relatively hard such that we need to incorporate some contextual information in the reasoning procedure.

Table 2: Ablation study for the depth unit setting with our lightweight model on nuScenes.

| $U$ (meters) | mAP | mATE | mASE | mAOE | mAAE | NDS |
|---|---|---|---|---|---|---|
| 5 | 0.298 | 0.79 | 0.266 | 0.563 | 0.164 | 0.371 |
| 10 | 0.303 | 0.775 | 0.265 | 0.548 | 0.164 | 0.376 |

Table 3: Ablation study for alternative depth division methods.

| Methods | Easy | Mod. | Hard |
|---|---|---|---|
| Log | 9.91 | 8.68 | 7.95 |
| Linear | 18.63 | 14.49 | 13.25 |
| Uniform Log | 8.62 | 13.48 | 13.28 |
| Uniform | **19.10** | **16.04** | **14.83** |

**Ablation Studies for Alternative Depth Division Methods** We also made ablation studies for alternative probabilistic depth settings, including the different settings for the depth unit $U$ and different division methods to bucket the depth value into intervals. First, Tab. 2 shows that more fine-grained division can not bring performance gains. As for the division methods, we test several alternatives as shown in Tab. 3, among which *Log* and *Linear* refer to the spacing-increasing discretization (SID) [8] and linear-increasing discretization (LID) [9], respectively. We directly take their split points and compute the depth estimation with Eqn. 1 in the main paper. In contrast, *Uniform Log* means that we take the split points that are uniformly distributed in the *log* space as the base to compute the depth estimation in the *log* space with Eqn. 1, and then apply the exponential transformation to get the final result. We can see that although the simplicity, our adopted uniform division method achieves the best performance. Note that this ablation study is conducted with $U = 10m$. There may be different conclusions if we exploit more fine-grained divisions or use classification and residual regression to implement the probabilistic depth estimation.

**Ablation Studies for Geometric Depth** Recall that we select three important factors for edge pruning and gating in the depth propagation graph. We also tried other alternatives for the distance score, including the height difference between 3D bottoms, the distance of 3D centers and our adopted 2D centers (Tab. 4). It can be observed that using the 2D centers yields the best performance. We conjecture that it is because the 3D criteria are based on the inaccurate depths such that they are less reliable than the disentangled 2D distance.

**Depth Error Analysis** We have validated the efficacy of our method in the main paper by comparing the detection performance of our method and the baseline, especially in terms of the improved

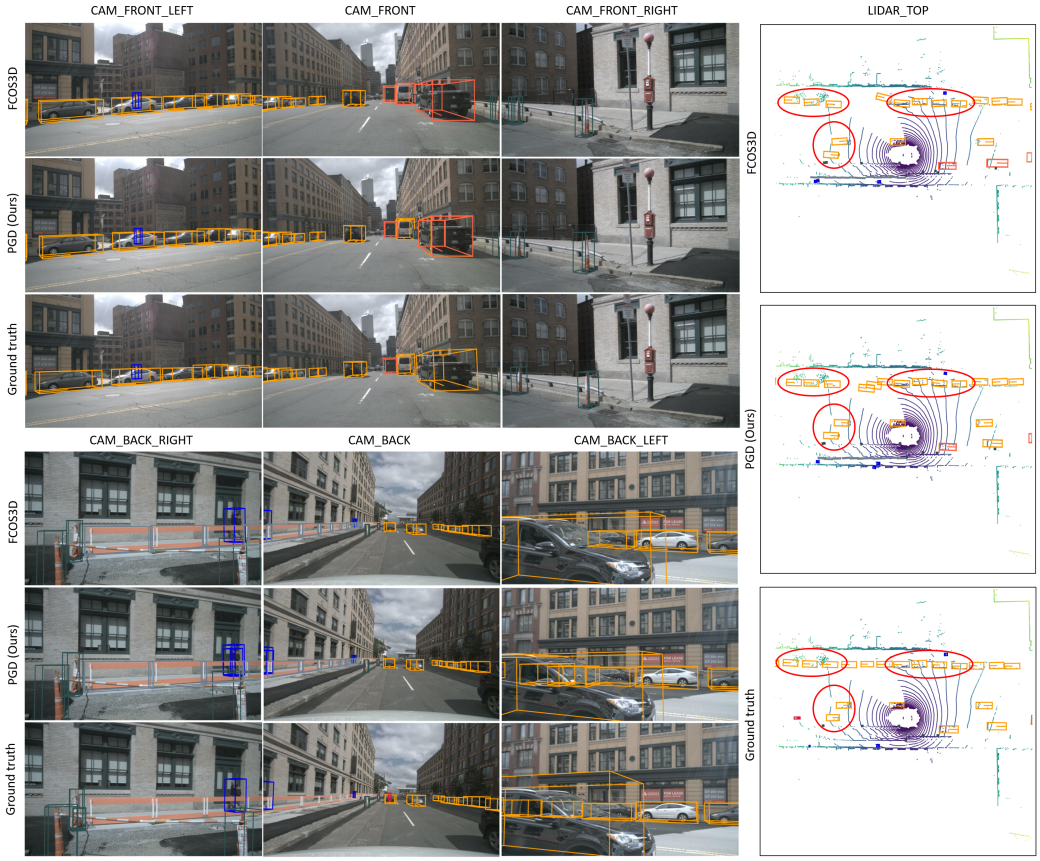


Figure 9: Qualitative analysis of detection results. 3D bounding box predictions are projected onto images from six different views and bird-view, respectively. Boxes from different categories are marked with different colors. We can see that the detection results of FCOS3D and PGD are both reasonable. However, from the bird-eye-view, the depth accuracy is remarkably improved by our method, especially for those objects marked with red circles.

Table 4: Ablation study for alternative distance scores in the edge gating scheme on KITTI.

| Method | $AP_{3D}$ IOU$\geq$ 0.7 | | | $AP_{3D}$ IOU$\geq$ 0.5 | | |
|---|---|---|---|---|---|---|
| | Easy | Mod. | Hard | Easy | Mod. | Hard |
| 3D bottoms | 15.18 | 11.96 | 10.72 | 46.27 | 37.99 | 33.09 |
| 3D centers | 21.04 | 16.07 | 14.89 | 47.03 | 37.58 | 32.97 |
| 2D centers | 21.36 | 16.60 | 15.60 | 50.57 | 39.78 | 34.18 |

Table 5: Depth error statistics for predictions having corresponding matching ground truths.

| Methods | Mean Abs. Error (m) ↓ | Mean Rel. Error ↓ |
|---|---|---|
| FCOS3D | 0.0528 | 4.27% |
| PGD (Ours) | 0.0483 | 3.63% |
| Rel. Delta | -8.5% | -15.0% |

mean average precision (mAP) and the mean translation error (mATE). Here we further prove its effectiveness with the depth error analysis. We make depth error statistics for the predictions (before NMS) which have corresponding ground truths on the KITTI validation set (Tab. 5). We can observe that our method significantly reduces the mean error of depth estimation, both on the absolute error and relative error ((Abs. and Rel. in Tab. 5).

### 3.3 Qualitative Analysis

Then we show some qualitative results on nuScenes in Fig. 9 by drawing the predicted 3D bounding boxes in the six-view images and the top-view point clouds. We compare the results predicted by our model and the baseline FCOS3D to demonstrate the improvements in terms of depth estimation intuitively. We can see that from the perspective of images, both detection results are appealing, especially for some small objects that are not labeled. For example, the barriers in the rear right camera are not labeled but detected by these two models. However, from the bird-eye-view, the depth accuracy of the two methods is notably different, especially for those objects marked with red circles: The accuracy is significantly improved by our proposed method. It is also in line with the quantitative results (the mATE is reduced remarkably) and further validates the efficacy of our method.

## References


[1] T. Wang, X. Zhu, J. Pang, and D. Lin. FCOS3D: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, 2021.

[2] T. Wang, C. He, Z. Wang, J. Shi, and D. Lin. Flava: Find, localize, adjust and verify to annotate lidar-based point clouds. In *Adjunct Publication of the 33rd Annual ACM Symposium on User Interface Software and Technology*, UIST '20 Adjunct, page 31–33, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450375153. doi:10.1145/3379350.3416176. URL https://doi.org/10.1145/3379350.3416176.

[3] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[4] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[5] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. *CoRR*, abs/1903.11027, 2019. URL http://arxiv.org/abs/1903.11027.

[6] A. Simonelli, S. R. R. Bulò, L. Porzi, M. López-Antequera, and P. Kontschieder. Disentangling monocular 3d object detection. In *IEEE International Conference on Computer Vision*, 2019.

[7] X. Zhou, D. Wang, and P. Krähenbühl. Objects as points. *CoRR*, abs/1904.07850, 2019. URL https://arxiv.org/abs/1904.07850.

[8] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao. Deep ordinal regression network for monocular depth estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[9] Y. Tang, S. Dorn, and C. Savani. Center3d: Center-based monocular 3d object detection with joint depth understanding. *CoRR*, abs/2005.13423, 2020. URL https://arxiv.org/abs/2005.13423.