DIGGING INTO OUTPUT REPRESENTATION FOR MONOCULAR 3D OBJECT DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Monocular 3D object detection aims to recognize and localize objects in 3D space from a single image. Recent researches have conducted remarkable advancements, while all of them follow a typical output representation in LiDAR-based 3D detection. However, in this paper, we argue that the existing discrete output representation is not suitable for monocular 3D detection. Specifically, monocular 3D detection has only two-dimensional information input while is required to output three-dimensional detections. This characteristic indicates that monocular 3D detection is inherently different from other typical detection tasks that have the same dimensional input and output. The dimension gap causes a large lower bound for the error of estimated depth. Therefore, we propose to reformulate the existing discrete output representation as a spatial probability distribution according to depth. This probability distribution considers the uncertainty caused by the absent depth dimension, allowing us to accurately and comprehensively represent objects in 3D space. Extensive experiments exhibit the superiority of our output representation. As a result, we have applied our method to 12 SOTA monocular 3D detectors, consistently boosting their average precision (AP) by $\sim$ **20% relative improvements**. The source code will be publicly available soon.

1 INTRODUCTION

Monocular 3D object detection is an important topic and has drawn much attention from the computer vision and autonomous driving community. It enables cars and robots to perceive the world in 3D using only a single camera. However, reasoning 3D box from monocular imagery is extremely challenging due to its inherent ill-posed nature. Towards boosting the accuracy, prior works have done much attempts, including utilizing estimated depth maps Wang et al. (2019); Ma et al. (2020); Reading et al. (2021), geometry natures Mousavian et al. (2017); Zhou et al. (2021); Zhang et al. (2021), and network designs Brazil & Liu (2019); Li et al. (2020).

All prior monocular works employ the typical output representation emerging in earlier detection tasks Zhou et al. (2019); Shi et al. (2020); Lang et al. (2019), *i.e.*, 2D box detections with corresponding 3D boxes, where the 3D boxes are regarded as final results. However, this discrete output representation ignores an inherent huge gap between monocular 3D detection and other detection tasks. As shown in Table 1, we summarize the input/output of different detection tasks and their dimensions in the source domain. For monocular 3D detection, it is required to reason high dimensional 3D boxes, while it has only low dimensional information input. This gap does not exist in the other two detection tasks, and our quantitative experiments prove that exactly the dimension gap results in low detection accuracy in monocular 3D detection.

Table 1: Differences in three detection tasks. "*Dim.*" in the table refers to the dimension.

Tasks	Input	Output	Input/Output Dim.	Dim. Gap
2D detection	2D image	2D boxes	2D/2D	✗
LiDAR-based 3D detection	3D point cloud	3D boxes	3D/3D	✗
Monocular 3D detection	2D image	3D boxes	2D/3D	✓

Please note, the *"2D"* and *"3D"* mentioned in the table refer to the data dimension from the source domain, but not the data representation. It is because changing the data representation (*e.g.*, range-view and pseudo-LiDAR) does not change the entropy.

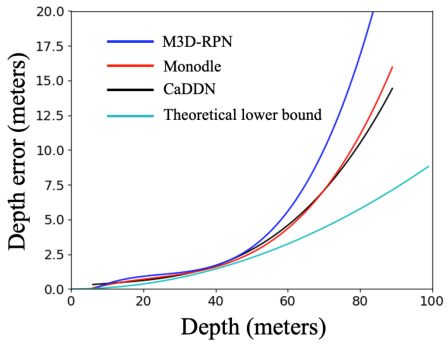

Figure 1: Depth error. We show mean depth errors of three SOTA monocular detectors in the given depth range and exhibit the theoretical lower bound.

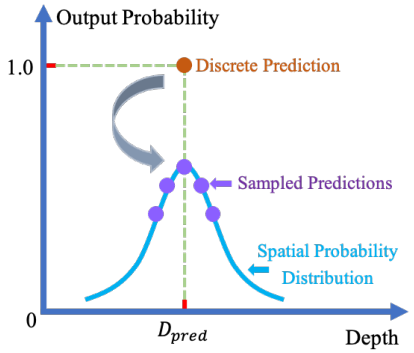

Figure 2: We convert a discrete output as a spatial probability distribution, and then sample points as detection candidates from the distribution.

The absent dimension is the depth, meaning that the network is forced to predict depth in 3D space from 2D visual features. To analysis the depth error brought by the dimension gap, we show error curves of recent SOTA monocular 3D detectors Brazil & Liu (2019); Ma et al. (2021); Reading et al. (2021) in Figure 1. The depth errors increase exponentially with the growth of depth. We also show a theoretical lower bound, which is quadratically increasing (See Section 3.2 for detailed derivation). Both depth errors in SOTA detectors and the theoretical lower bound indicate that estimated depths cannot be accurate for objects which are not close. Taking KITTI dataset as an example, the theoretical lower bound for depth error is around 1.48 meters and 3.33 meters for objects 40 meters and 60 meters away, respectively. Although more and more advancements are shown in monocular 3D detection, the lower bound is hard to be broken.

From the perspectives of the inherent dimension gap and the resulting large depth errors, we argue that the existing representation of a discrete depth prediction is suboptimal for monocular 3D detection. Large depth errors mean that the predicted depths have large uncertainty, and the resulting discrete 3D box predictions cannot accurately and comprehensively represent the object status in 3D space. Therefore, in this paper we aim to reformulate the output representation for monocular 3D detection to take the inherent dimension gap and depth uncertainty into consideration.

Our reformulation consists of two steps: First, as shown in Figure 2, we transform each discrete detection output as a spatial probability distribution using the normal distribution, where the standard deviation increases with the growth of depth. Second, we sample multiple 3D boxes from the spatial distribution while considering the depth uncertainty and regard them as new detection results. In other words, instead of outputting only one discrete 3D box, we transform the 3D box to a continuous spatial probability distribution in 3D space and then produce more predictions via sampling. Our method considers the underlying absent depth dimension, utilizing depth uncertainty in output representation for monocular 3D detection, therefore consistently and considerably boosting the performance for most detectors.

We summarize our main contributions as below:

- We rethink the underlying mechanism of monocular 3D object detection, arguing that it is inherently different from other detection tasks. Based on our analysis, we reformulate the discrete output representation as a spatial probability distribution, which is more reasonable for monocular 3D detection.
- We have applied our method to 12 recent SOTA monocular 3D detectors, consistently boosting their average precision (AP) by $\sim$ **20% relative improvements**. It is worthy to note that our method can be easily adapted to any monocular 3D detector, which does not bring extra costs.

2 RELATED WORK

2.1 LIDAR 3D OBJECT DETECTION

Current LiDAR-based methods Zheng et al. (2021); Shi et al. (2019b); Lang et al. (2019); Shi & Rajkumar (2020); Yang et al. (2020); Shi et al. (2020); He et al. (2020) are the main steam and widely deployed in the industry due to their high accuracy. Generally speaking, LiDAR-based methods can be categorized into two types of methods: voxel-based methods Zhou & Tuzel (2018) and point-based methods Qi et al. (2018); Shi et al. (2019a) . For voxel-based methods, they typically divide the raw point cloud into voxel grids, extracting a unified feature representation for each voxel. In this way, the irregular and unordered point clouds are formed as ordered and CNN-friendly data representations, *i.e.*, voxel grids. Therefore CNNs can easily extract features from such voxels and then regress required 3D box parameters. On the other hand, point-based methods directly extract features from the raw point cloud using light fully connected networks Qi et al. (2017a;b). This manner does not damage the 3D information, while usually cost more time due to a large number of points. Both types of methods can be further divided into one-stage Yan et al. (2018); Lang et al. (2019); Zheng et al. (2020) and two-stage methods Yang et al. (2019); Shi et al. (2020), where the two-stage methods usually use RoI features to refine initial detections in the first stage, thus producing better results. In sum, thanks to the precise 3D measurements, LiDAR-based methods take the predominant place in 3D detection in terms of detection accuracy.

2.2 MONOCULAR 3D OBJECT DETECTION

Despite the high accuracy, LiDAR-based methods have disadvantages of heavy costs. In light of the low deployment overhead, monocular-based methods have been popular. Early methods usually use projection geometry constraints or semantic prior. Mono3d Chen et al. (2016) first proposes an energy minimization approach, to place object candidates on the ground plane, then score each candidate via several intuitive potentials encoding semantic segmentation, contextual information, size and location priors and object shape. This pipeline is rather complex, and Deep3DBox Mousavian et al. (2017) introduces a simple yet effective method. With the help of a mature 2D detector, it only needs to regress the orientation and dimension of objects, and the most difficult object 3D location is estimated by 2D-3D projection constraints. RoI-10D Manhardt et al. (2019) introduces an end-to-end monocular 3D object detection method, with a novel loss formulation by lifting 2D detection, orientation, and scale estimation into 3D space. Later methods recognize the importance of instance depth estimation. For instance, M3D-RPN Brazil & Liu (2019) designs depth-aware convolutional layers which enable location specific feature development. Following this line of thought, D4LCN Ding et al. (2020) uses estimated depth maps to generate dynamic convolution kernels to extract features in different 3D locations. And more recently, CaDDN Reading et al. (2021) utilizes estimated categorical pixel-wise depth distribution to project features to the 3D space. Another stream of using estimated depth maps is to convert the input data representation. Pseudo-LiDAR Wang et al. (2019) converts depth maps estimated by an off-the-shelf depth estimator to point clouds, mimicking the real LiDAR signal. PatchNet Ma et al. (2020) directly concatenate the RGB image patch with 3D coordinates of transformed depth, *i.e.*, x, y, z, and obtain better results. Some methods explore the main challenge in monocular 3D detection. Monodle Ma et al. (2021) attempts to analyze which type of error accounts for the poor accuracy, and they find the localization error is the main reason. However, they do not consider the influence of depth on other parameters in location, and our experiments show that the instance depth in the location is the main reason for low detection rates.

2.3 OUTPUT REPRESENTATION IN 2D AND 3D DETECTION

Current 2D detection Ren et al. (2015); Redmon & Farhadi (2018) commonly adopts the same output data representation, *i.e.*, the 2D box coordinates and corresponding confidence, where the confidence usually denotes the classification score. For 3D detection, existing methods use a similar representation. Typically, for LiDAR-based 3D detectors Lang et al. (2019); Shi et al. (2020), the output 3D box is parameterized by location (x, y, z), dimension (h, w, l), orientation (θ), and confidence (C). Most monocular-based 3D detectors Brazil & Liu (2019); Li et al. (2020) follow this output representation. Many monocular-based 3D detectors Mousavian et al. (2017); Wang et al. (2019); Ma et al. (2020) predict independent 2D boxes, lifting each 2D box prediction to a 3D box by predicting required 3D parameters. Therefore, the final confidence in monocular 3D detectors refers to

the 2D classification score from 2D box Ma et al. (2020) or the score combing 2D score with the difficulty of lifting process from 2D to 3D Simonelli et al. (2019). Current monocular methods all do not consider the depth uncertainty brought by the dimension gap in the output representation.

3 What Makes Monocular 3D Detection Challenging?

3.1 Dilemma in Monocular 3D Detection

Most prior works do not pay much attention to the dimension gap between input and output. Considering the other two maturer detection tasks: 2D detection and LiDAR-based 3D detection, they all have the same dimensional information with respect to input and output, namely, 2D to 2D for 2D detection and 3D to 3D for LiDAR-based 3D detection. Nevertheless, monocular 3D detection has only 2D information while is required to output precise 3D information. The absent dimension is exactly the depth, which is the reason why predicting precise depth is very challenging.

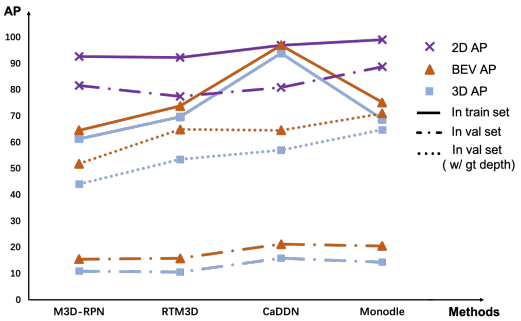

Figure 3: The performance gap in monocular 3D detection. We show the performance of different SOTA detectors on *train* and *val* set, respectively. We can see that when eliminating the influence of depth (replace depth prediction with the ground-truth depth), the 3D detection performance (including BEV and 3D AP) is largely boosted, even showing the similar tendency compared to 2D detection. Note that when using the ground-truth depth in *val* set, we adjust the x, y in the location to fit the new depth.

Benefit from the deep learning technology, a network is allowed to be trained using massive labeled data, to learn the underlying mechanism for the target task and then conduct predictions on unseen data. Unfortunately, on the physical level, it is impossible to reason accurate depth from a single image due to the dimension gap, therefore the generalization ability of monocular depth estimation is largely limited. We conduct experiments to demonstrate this point. As shown in Figure 3, we show the performance of different state-of-the-art monocular detectors on *train* and *val* set. We can observe that the 3D detection accuracy (including BEV (bird's-eye-view) and 3D AP) on *train* set is high (higher than 60 AP) while the accuracy on *val* set data is extremely low (lower than 20 AP). As expected, when removing the influence of estimated depth (replace depth prediction with the ground-truth depth), the 3D detection performance is largely boosted. The improved 3D detection accuracy on *val* set is comparable to the accuracy on *train* set. It indicates that the dimension gap is the main obstacle for monocular 3D detection.

3.2 Monocular Depth Estimation Error Lower Bound

In this section, we discuss the error lower bound of estimated depth for monocular imagery. The camera captures and represents the current scene using an image, quantifying the scene with pixels. Assume that the calibrated camera system has the intrinsics parameters $A = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$. We denote all points on the image and in the 3D space as Q and P, where $Q \in R^2$ and $P \in R^3$, respectively. Two different 3D points may be projected into one pixel, thus we use the depth offset within one pixel to denote the depth error lower bound. This error also can be regarded as a quantified error. We have two points: $p_1 = [x_1, y_1, z_1]^T$, $p_2 = [x_1, y_2, z_2]^T$, $z_2 > z_1$, and $p_1, p_2 \in P$. We know that $Q = AP$ according to camera projection. The corresponding points onto the image are denoted by $q_1 = [u_1, v_1]^T$, $q_2 = [u_2, v_2]^T$, $q_1, q_2 \in Q$. Such two points have one pixel offset in terms of the projections on the image. To calculate the depth error, we follow a basic assumption in the autonomous driving scenario, namely, the ground plane is rather flat. It means that $y_1 \approx y_2 = c$ as the inclined angle for the ground plane is low, where c denotes the distance from the camera to

the ground plane. Thus we use the vertical pixel offset to represent the depth error, we obtain:

$$v_2 - v_1 = \frac{f_y * y_2 + c_y * z_2}{z_2} - \frac{f_y * y_1 + c_y * z_1}{z_1} = 1 \tag{1}$$

We have:

$$\delta_z = z_2 - z_1 = \frac{{z_1}^2}{f_y c - z_1} < \frac{{z_1}^2}{f_y c} \tag{2}$$

We can see that this theoretical depth error lower bound increases quadratically with the growth of depth, which indicates that the depth estimation by monocular imagery has an immanent drawback. Thus monocular 3D detection always shows poor accuracy for far objects.

4 Reformulate Output Representation for Monocular 3D Detection

4.1 Why not Using Previous Detection Output Representation?

Considering the dimension gap in monocular 3D detection, *i.e.*, the absent depth dimension in the lifting process from 2D to 3D, we rethink the detection output representation, with raising a question: the discrete representation adopted by previous works is indeed suitable? Unfortunately, we give a negative answer. We list our main reasons as below: **(i)** Different dimensional information for input. Given only 2D information, the monocular 3D detection task is required to output precise 3D information. The discrete and unique output cannot precisely reflect the uncertainty in the lifted prediction process. **(ii)** Extensive 3D outdoor space. It is very challenging to reason a discrete yet accurate 3D box in the extremely extensive 3D outdoor space. The ill-posed monocular imagery further enhances the difficulty of this challenge. **(iii)** The high reliance on safety in the autonomous driving scenario. To avoid collision with other obstacles and better planning, the ego-car/robot should detect as many precise locations of objects as possible in the current scene, *i.e.*, pursuing a higher recall, while current discrete representation is hard to achieve this goal.

4.2 Spatial Probability Distribution in Output Representation

Based on the above analysis, we propose to reformulate the output representation for monocular 3D detection. Focusing on handling the absent depth dimension, we transform the discrete detection output as a spatial probability distribution in the depth range. Considering the known fact: since the precise instance depth is unachievable, we can use the spatial probability distribution to more comprehensively represent the 3D object. This representation also provides more valuable information such as location uncertainty nearby the object for downstream tasks, *e.g.*, tracking and planning.

Specifically, for a discrete monocular 3D object prediction parameterized by the location (x, y, z), dimension (h, w, l), orientation (θ), and confidence (C), we first convert the depth z to the normal distribution $N(z, \sigma)$. Since this probability distribution is to reflect the relative uncertainty of the depth prediction in 3D space, we use the relative probability to represent the depth uncertainty and thus the final depth confidence is as follows:

$$t(s) = e^{-\frac{(s-z)^2}{\sigma^2}}, \qquad \sigma = e^{\frac{z}{\lambda}} \tag{3}$$

where s denotes any depth and $t(s)$ refers to the relative depth confidence. With the growth of depth, the standard deviation in the normal distribution is becoming larger as the depth is more and more difficult to be predicted. Combing with the original confidence (typically the classification), the final confidence for the object at depth s is: $C_s = C \cdot t(s)$. Also, due to the projection relationship, the location x, y should be changed to fit the new depth s. Therefore the location of object at new depth s is $(\frac{x}{z}s, \frac{y}{z}s, s)$. Other parameters (dimension and orientation) do not change. We term this spatial probability distribution for location the location distribution.

The location distribution has serval advantages. First, it is naturally suitable for monocular 3D detectors as it directly expresses the uncertainty of the predicted depth into the output. Second, it is more comprehensively and accurately to describe the obstacles status, while prior methods using the discrete output representation can be overconfident on a less accurate prediction. Third, this output representation is flexible, which is also compatible with the previous representation (when λ in Equation 3 is set to $-\infty$).

4.3 SAMPLING STRATEGY

By using the location distribution for each discrete original output, we have a series of probabilities representation for objects. Then, we sample locations from the location distribution as new detection results, to evaluate the results. We propose two sampling strategies and show them in Figure 4.

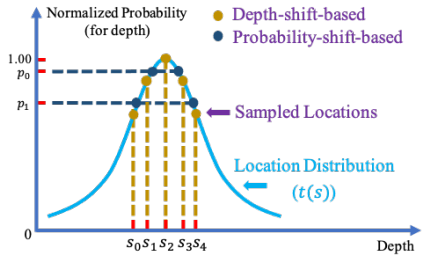

Figure 4: Two sampling strategies in the location distribution.

• **Depth-shift-based Sampling.** We first define a depth shift set prior and then use the depth shift from the set for each original detection output. Each depth shift can associate with one location in the location distribution. For the object with the original depth z, the new detection results are:

$$Loc_{new} = f(z + d_s), \qquad d_s \in D_s \tag{4}$$

As shown in Equation 4, Loc_{new} denotes new locations and D_s refers to the depth shift set, and f refers to the function of sampling locations according to depth.

• **Probability-shift-based Sampling.** Similar to depth-shift-based sampling, we use a prior probability set to sample locations.

$$Loc_{new} = g(p_s), \qquad p_s \in P_s \tag{5}$$

As shown in Equation 5, P_s refers to the probability shift set, and g refers to the function of sampling locations according to the relative depth probability. Our experiments show that both sampling strategies bring significant improvements for the original detector.

5 EXPERIMENTS

5.1 IMPLEMENTATION DETAILS

For the λ in Equation 3, we use 80 for KITTI Geiger et al. (2012) and 160 for Waymo Sun et al. (2020) as Waymo dataset covers larger depth ranges. We use the set $[\pm 2, \pm 1, \pm 0.5, 0]$ meters for depth-interval-based sampling and the set $[0.7, 0.8, 0.9, 1.0]$ for probability-shift-based sampling. The depth-interval-based sampling strategy is employed by default. We re-implement the baseline detectors using officially released codes for validation. We employ CaDDN Reading et al. (2021) for most ablation studies. Also, considering that close objects are accurate enough in terms of depth estimation, we do not transform the output representation for objects within 10 meters.

5.2 DATASET AND METRICS

We conduct experiments in KITTI Geiger et al. (2012) and Waymo Sun et al. (2020) dataset. KITTI dataset is the widely employed benchmark for monocular 3D detection. Specifically, KITTI provided 7481 samples for training and 7518 samples for testing, where the training set is publicly available and labels in the test keep secret. To make fair comparisons, we adopt the commonly used training/validation dataset split introduced in Chen et al. (2017), which divides the public available data into a new training set including 3712 samples and a validation set including 3769 samples. We conduct experiments on the KITTI validation set and official test set under two core tasks: bird's eye view (BEV) and 3D object detection in three difficulties. Difficulties of objects are subdivided into easy, moderate, and hard in terms of the occlusion level, truncation, and bounding box height. We provide performances under AP_{40} metrics Simonelli et al. (2019) to evaluate the proposed method.

Recently, some large public datasets for autonomous driving are released, where the Waymo Open Dataset Sun et al. (2020) is a typical one. Waymo dataset consists of 798 training sequences and 202 validation sequences. Different from KITTI, it provides 3D box labels in the 360° field of view (FOV), while we only use the front view for the monocular task, and we use the same data processing strategy proposed in CaDDN Reading et al. (2021).

Table 2: Performance of using our representation on different SOTA monocular detectors. We only change the output representation. All methods are evaluated on KITTI *val* set with metric $AP|_{R_{40}}$.

Approaches	AP_{BEV} /AP_{3D} (IoU=0.7)$\|_{R_{40}}$ Easy	Moderate	Hard	Relative Improvements
MonoGRNet Qin et al. (2019)	19.58/11.85	12.73/7.52	10.08/5.73	
MonoGRNet+Ours	**23.95/15.63**	**17.11/10.70**	**13.51/8.45**	22%-47%
Improvements	+4.37/+3.78	+4.38/+3.18	+3.43/+2.72	
M3D-RPN Brazil & Liu (2019)	20.57/14.36	15.46/10.95	11.76/8.58	
M3D-RPN+Ours	**23.86/17.42**	**19.15/13.70**	**15.19/10.87**	16%-29%
Improvements	+3.29/+3.06	+3.69/+2.75	+3.43/+2.29	
Pseudo-LiDAR Wang et al. (2019)	37.77/25.19	21.31/12.72	17.92/10.22	
Pseudo-LiDAR +Ours	**41.00/27.98**	**25.67/15.73**	**21.87/13.13**	9%-28%
Improvements	+3.23/+2.79	+4.36/+3.01	+3.95/+2.91	
RTM3D Li et al. (2020)	20.72/13.02	15.78/10.60	13.84/9.25	
RTM3D+Ours	**26.99/18.19**	**22.10/15.70**	**20.28/14.17**	30%-53%
Improvements	+6.27/+5.17	+6.32/+5.10	+6.44/+4.92	
D4LCN Ding et al. (2020)	31.82/22.85	22.43/16.02	17.05/12.21	
D4LCN +Ours	**35.81/26.48**	**26.70/19.59**	**21.09/15.34**	13%-26%
Improvements	+3.99/+3.63	+4.27/+3.57	+4.04/+3.13	
KM3D Li & Zhao (2021)	25.07/16.01	17.85/11.68	15.60/10.57	
KM3D +Ours	**30.64/21.46**	**24.13/17.27**	**21.74/15.59**	22%-48%
Improvements	+5.57/+5.45	+6.28/+5.59	+6.14/+5.02	
PatchNet Ma et al. (2020)	43.97/32.56	25.43/17.71	20.73/13.98	
PatchNet +Ours	**46.72/36.69**	**29.79/21.19**	**25.65/17.90**	6%-28%
Improvements	+2.75/+4.13	+4.36/+3.48	+4.92/+3.92	
GrooMeD-NMS Kumar et al. (2021)	27.25/19.60	19.65/14.28	15.87/11.25	
GrooMeD-NMS +Ours	**32.19/23.94**	**25.77/19.07**	**20.54/14.85**	18%-32%
Improvements	+4.94/+4.34	+6.12/+4.79	+4.67/+3.60	
MonoFlex Zhang et al. (2021)	28.28/20.02	21.56/15.19	18.79/12.95	
MonoFlex +Ours	**33.57/24.83**	**27.48/20.62**	**24.48/18.44**	19%-42%
Improvements	+5.29/+4.81	+5.92/+5.43	+5.69/+5.49	
CaDDN Reading et al. (2021)	30.98/23.19	21.18/15.84	19.14/13.42	
CaDDN +Ours	**35.66/27.25**	**26.97/20.23**	**24.89/18.13**	15%-35%
Improvements	+4.68/+4.06	+5.79/+4.39	+5.75/+4.71	
Monodle Ma et al. (2021)	25.26/17.37	20.51/14.34	17.93/12.83	
Monodle +Ours	**29.47/22.52**	**25.23/19.57**	**22.52/17.29**	17%-35%
Improvements	+4.21/+5.15	+4.72/+5.23	+4.59/+4.46	
LPCG Peng et al. (2021)	40.24/31.15	30.55/23.42	27.32/20.60	
LPCG +Ours	**44.93/36.07**	**36.63/29.56**	**33.72/26.97**	13%-31%
Improvements	+4.69/+4.92	+6.08/+6.14	+6.40/+6.37	

Table 3: Comparisons on Waymo. "*Rel. Imp.*" in the table refers to relative improvements.

Represen-tations	3D mAP Overall	0−30m	30−50m	50m−∞	3D mAPH Overall	0−30m	30−50m	50m−∞
Previous	5.19	19.08	2.26	0.15	5.00	18.52	2.14	0.14
Ours	**6.66**	**23.27**	**3.67**	**0.25**	**6.40**	**22.52**	**3.45**	**0.23**
Rel. Imp.	28.33%	21.96%	62.39%	66.67%	28.00%	21.60%	61.21%	64.29%

5.3 Comparisons on KITTI and Waymo Dataset

As shown in Table 2, we provide the results on KITTI *val* set on 12 SOTA monocular 3D detectors. We can see that the performance of original methods is largely boosted by employing our output

representation. For instance, we improve the AP of M3D-RPN Brazil & Liu (2019) by 15.99%-29.17% relative improvements and boost the AP of MonoFlex Zhang et al. (2021) by 18.71%-42.39% relative improvements, which are rather impressive results. The consistent improvements on most SOTA detectors demonstrate the effectiveness and robustness of our method. We show results on KITTI test set in Table 4, and we set a new state-of-the-art with a considerable margin.

To further demonstrate the robustness of the proposed method, we conduct experiments on Waymo dataset, which is a new large dataset for the research of autonomous driving. We exhibit the results in Table 3, we can see that our method brings significant improvements.

Table 4: Comparisons on KITTI testing set. We use LPCG as the baseline detector, and our method outperforms other methods with a considerable margin.

Approaches	AP_{BEV} (IoU=0.7)$\|_{R_{40}}$			AP_{3D} (IoU=0.7)$\|_{R_{40}}$		
	Easy	Moderate	Hard	Easy	Moderate	Hard
MonoGRNet Qin et al. (2019)	18.19	11.17	8.73	15.74	9.61	4.25
MonoPSR Ku et al. (2019)	18.33	12.58	9.91	10.76	7.25	5.85
AM3D Ma et al. (2019)	25.03	17.32	14.91	16.50	10.74	9.52
M3D-RPN Brazil & Liu (2019)	21.02	13.67	10.23	14.76	9.71	7.42
MonoPair Chen et al. (2020)	19.28	14.83	12.89	13.04	9.99	8.65
D4LCN Ding et al. (2020)	22.51	16.02	12.55	16.65	11.72	9.51
RTM3D Li et al. (2020)	19.17	14.20	11.99	14.41	10.34	8.77
PatchNet Ma et al. (2020)	22.97	16.86	14.97	15.68	11.12	10.17
Kinematic3D Brazil et al. (2020)	26.69	17.52	13.10	19.07	12.72	9.17
Neighbor-Vote Chu et al. (2021)	27.39	18.65	16.54	15.57	9.90	8.89
MonoRUn Chen et al. (2021)	27.94	17.34	15.24	19.65	12.30	10.58
MonoRCNN Shi et al. (2021)	25.48	18.11	14.10	18.36	12.65	10.03
DDMP-3D Wang et al. (2021)	28.08	17.89	13.44	19.71	12.78	9.80
Monodle Ma et al. (2021)	24.79	18.89	16.00	17.23	12.26	10.29
CaDDN Reading et al. (2021)	27.94	18.91	17.19	19.17	13.41	11.46
Ground-Aware Liu et al. (2021a)	29.81	17.98	13.08	21.65	13.25	9.91
GrooMeD-NMS Kumar et al. (2021)	26.19	18.27	14.05	18.10	12.32	9.65
MonoEF Zhou et al. (2021)	29.03	19.70	17.26	21.29	13.87	11.71
MonoFlex Zhang et al. (2021)	28.23	19.75	16.89	19.94	13.89	12.07
AutoShape Liu et al. (2021b)	30.66	20.08	15.95	22.47	14.17	11.36
DD3D Park et al. (2021)	30.98	22.56	20.03	23.22	16.34	14.20
LPCG Peng et al. (2021)	35.96	24.81	21.86	25.56	17.80	15.38
Ours	**39.74**	**28.84**	**26.08**	**29.15**	**21.24**	**19.18**
Improvements	+3.78	+4.03	+4.22	+3.59	+3.44	+3.80

5.4 Effectiveness Analysis

Our output representation shows promising improvements in terms of performance numbers. Here we give a more intuitive interpretation for the improvements via the P-R (precision-recall) curve. In Figure 5, we show P-R curves of a monocular 3D detector that uses different output representations. We can see that our method does not have impacts on the high precision and low recall regions, while works on low precision and high recall regions. Therefore, our method mostly boosts the recall to improve the performance of the monocular 3D detector, consequently improving the safety of the system equipped with monocular 3D detectors.

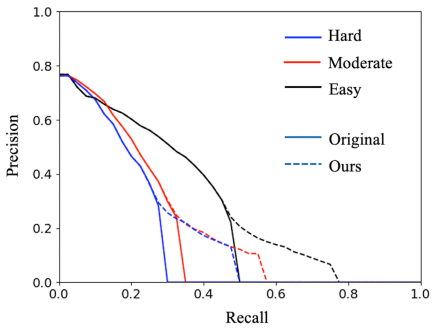

Figure 5: P-R curves of using previous representation and ours on KITTI *test* set. Solid lines: previous; dashed lines: ours.

5.5 Ablation Studies

We perform ablation studies to investigate the impact of each component on our method.

Table 5: Ablation for different sampling number on the location distribution.

Sampling Number	AP_{BEV} /AP_{3D} (IoU=0.7)$\|_{R_{40}}$ Easy	Moderate	Hard
Baseline(1)	30.98/23.19	21.18/15.84	19.14/13.42
3	34.54/26.35	25.51/19.15	21.18/16.75
5	35.43/27.17	26.63/19.92	24.18/17.99
7	**35.68/27.25**	**26.99/20.24**	**24.91/18.14**
9	35.66/27.25	26.97/20.23	24.89/18.13

Table 6: Ablation for depth uncertainty. "*Un.*" in the table refers to uncertainty and "*L-D*" denotes location distribution.

Un.	L-D	AP_{BEV} /AP_{3D} (IoU=0.7)$\|_{R_{40}}$ Easy	Moderate	Hard
		30.98/23.19	21.18/15.84	19.14/13.42
	✓	9.74/8.34	9.79/7.86	9.74/7.38
✓	✓	**35.68/27.25**	**26.99/20.24**	**24.91/18.14**

• **Impact of Sampling Numbers.** When sampling locations from the location distribution, the sampling number also has impacts on the accuracy. We show the results in Table 5, and we can know that a proper sampling number is preferred.

• **Impact of Depth Uncertainty.** In Table 6, we show the influence of depth uncertainty in the location distribution. If the confidence of sampled locations are not weighted by the uncertainty, the performance of the original method is largely downgraded, demonstrating the importance of depth uncertainty in monocular 3D detection.

• **Impact of Sampling Strategy.** We use the proposed two sampling strategies, namely, depth-shift-based and probability-shift-based. As shown in Table 7, both two sampling strategies bring significant improvements for the original method, which demonstrate the effectiveness of our method.

Table 7: Ablation for sampling strategies.

Sampling Strategies	AP_{BEV} /AP_{3D} (IoU=0.7)$\|_{R_{40}}$ Easy	Moderate	Hard
Probability shift	**35.76/27.56**	26.93/**20.45**	24.25/**18.29**
Depth shift	35.68/27.25	**26.99**/20.24	**24.91**/18.14

• **Impact of Location Distribution.** In this paper we only transform spatial probability for estimated depth. To make a comprehensive comparison, we also apply this transformation on other location parameters, *i.e.*, x and y. We show the results in Table 8. We can see that imposing probability transformation into other parameters which can be well inferred by known dimension is unnecessary (x and y are usually denoted by the projection on the image and then recovered by depth).

5.6 Performance on Pedestrian and Cyclist

We also perform experiments on other categories., and provide the results in Table 9. We can see that our method also boosts the performance for other categories. It is worthy to note that the improvements on the pedestrian are not impressive like other categories. It is because the orientation estimation is another main challenge for the pedestrian whose shape and appearance are deformable.

Table 8: Ablation for probability distribution.

Probability Distribution	AP_{BEV} /AP_{3D} (IoU=0.7)$\|_{R_{40}}$ Easy	Moderate	Hard
$None$	30.98/23.19	21.18/15.84	19.14/13.42
$xy\ \ only$	30.47/22.91	20.93/15.48	18.21/13.20
$Depth + xy$	35.20/26.92	26.02/19.52	23.73/17.37
$Depth\ \ only$	**35.68/27.25**	**26.99/20.24**	**24.91/18.14**

Table 9: Performance on other categories.

Representations	Categories	AP_{BEV} /AP_{3D} (IoU=0.5)$\|_{R_{40}}$ Easy	Moderate	Hard
Previous	Pedestrian	13.78/11.64	10.25/8.80	8.07/6.74
Ours		**14.23/12.00**	**10.78/8.87**	**8.61/7.04**
Previous	Cyclist	2.86/2.74	1.65/1.39	1.25/1.21
Ours		**4.94/4.50**	**2.56/2.30**	**2.07/1.93**

6 Conclusion

In this paper, we review previous detection tasks, argue that the monocular 3D detection task is inherently different from other tasks. For monocular 3D detection, it lacks the depth dimension, thus performing worse on unseen data. We further use a depth error lower bound for monocular imagery to demonstrate this point. Therefore we propose to reformulate the previous discrete output representation as the spatial probability distribution to take depth estimation uncertainty into consideration. We also propose two sampling strategies to sample locations from location distribution. As a result, experiments exhibit that our output representation brings very promising improvements for most SOTA detectors. Additionally, considering the inherent depth uncertainty, we can use some network designs to further boosting the detection accuracy in future work.

REFERENCES

Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9287–9296, 2019.

Garrick Brazil, Gerard Pons-Moll, Xiaoming Liu, and Bernt Schiele. Kinematic 3d object detection in monocular video. In *European Conference on Computer Vision*, pp. 135–152. Springer, 2020.

Hansheng Chen, Yuyao Huang, Wei Tian, Zhong Gao, and Lu Xiong. Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10379–10388, 2021.

Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2147–2156, 2016.

Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals using stereo imagery for accurate object class detection. *IEEE transactions on pattern analysis and machine intelligence*, 40(5):1259–1272, 2017.

Yongjian Chen, Lei Tai, Kai Sun, and Mingyang Li. Monopair: Monocular 3d object detection using pairwise spatial relationships. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12093–12102, 2020.

Xiaomeng Chu, Jiajun Deng, Yao Li, Zhenxun Yuan, Yanyong Zhang, Jianmin Ji, and Yu Zhang. Neighbor-vote: Improving monocular 3d object detection through neighbor distance voting. *arXiv preprint arXiv:2107.02493*, 2021.

Mingyu Ding, Yuqi Huo, Hongwei Yi, Zhe Wang, Jianping Shi, Zhiwu Lu, and Ping Luo. Learning depth-guided convolutions for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11672–11681, 2020.

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361. IEEE, 2012.

Chenhang He, Hui Zeng, Jianqiang Huang, Xian-Sheng Hua, and Lei Zhang. Structure aware single-stage 3d object detection from point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11873–11882, 2020.

Jason Ku, Alex D Pon, and Steven L Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11867–11876, 2019.

Abhinav Kumar, Garrick Brazil, and Xiaoming Liu. Groomed-nms: Grouped mathematically differentiable nms for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8973–8983, 2021.

Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12697–12705, 2019.

Peixuan Li and Huaici Zhao. Monocular 3d detection with geometric constraint embedding and semi-supervised training. *IEEE Robotics and Automation Letters*, 6(3):5565–5572, 2021.

Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. *arXiv preprint arXiv:2001.03343*, 2020.

Yuxuan Liu, Yuan Yixuan, and Ming Liu. Ground-aware monocular 3d object detection for autonomous driving. *IEEE Robotics and Automation Letters*, 6(2):919–926, 2021a.

Zongdai Liu, Dingfu Zhou, Feixiang Lu, Jin Fang, and Liangjun Zhang. Autoshape: Real-time shape-aware monocular 3d object detection. *arXiv preprint arXiv:2108.11127*, 2021b.

Xinzhu Ma, Zhihui Wang, Haojie Li, Pengbo Zhang, Wanli Ouyang, and Xin Fan. Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 6851–6860, 2019.

Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. *arXiv preprint arXiv:2008.04582*, 2020.

Xinzhu Ma, Yinmin Zhang, Dan Xu, Dongzhan Zhou, Shuai Yi, Haojie Li, and Wanli Ouyang. Delving into localization errors for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4721–4730, 2021.

Fabian Manhardt, Wadim Kehl, and Adrien Gaidon. Roi-10d: Monocular lifting of 2d detection to 6d pose and metric shape. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2069–2078, 2019.

Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimtion using deep learning and geometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7074–7082, 2017.

Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? *arXiv preprint arXiv:2108.06417*, 2021.

Liang Peng, Fei Liu, Zhengxu Yu, Senbo Yan, Dan Deng, and Deng Cai. Lidar point cloud guided monocular 3d object detection. *arXiv preprint arXiv:2104.09035*, 2021.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017a.

Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017b.

Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 918–927, 2018.

Zengyi Qin, Jinglu Wang, and Yan Lu. Monogrnet: A geometric reasoning network for monocular 3d object localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 8851–8858, 2019.

Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8555–8564, 2021.

Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. 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, 2019a.

Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *arXiv preprint arXiv:1907.03670*, 2019b.

Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. 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, 2020.

Weijing Shi and Raj Rajkumar. 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, 2020.

Xuepeng Shi, Qi Ye, Xiaozhi Chen, Chuangrong Chen, Zhixiang Chen, and Tae-Kyun Kim. Geometry-based distance decomposition for monocular 3d object detection. *arXiv preprint arXiv:2104.03775*, 2021.

Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1991–1999, 2019.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2446–2454, 2020.

Li Wang, Liang Du, Xiaoqing Ye, Yanwei Fu, Guodong Guo, Xiangyang Xue, Jianfeng Feng, and Li Zhang. Depth-conditioned dynamic message propagation for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 454–463, 2021.

Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8445–8453, 2019.

Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

Zetong Yang, Yanan Sun, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Std: Sparse-to-dense 3d object detector for point cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1951–1960, 2019.

Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11040–11048, 2020.

Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3289–3298, 2021.

Wu Zheng, Weiliang Tang, Sijin Chen, Li Jiang, and Chi-Wing Fu. Cia-ssd: Confident iou-aware single-stage object detector from point cloud. *arXiv preprint arXiv:2012.03015*, 2020.

Wu Zheng, Weiliang Tang, Li Jiang, and Chi-Wing Fu. Se-ssd: Self-ensembling single-stage object detector from point cloud. *arXiv preprint arXiv:2104.09804*, 2021.

Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.

Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4490–4499, 2018.

Yunsong Zhou, Yuan He, Hongzi Zhu, Cheng Wang, Hongyang Li, and Qinhong Jiang. Monocular 3d object detection: An extrinsic parameter free approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7556–7566, 2021.