The Devil is in the Quality: Exploring Informative Samples for Semi-Supervised Monocular 3D Object Detection

Zhipeng Zhang[1*], Zhenyu Li[2*], Hanshi Wang[3,4*], Yuan He[1], Ke Wang[1] and Heng Fan[5]

***Abstract*— This paper tackles the challenging problem of semi-supervised monocular 3D object detection with a general framework. In specific, having observed that the bottleneck of this task lies in lacking reliable and informative samples from unlabeled data for detector learning, we introduce a novel simple yet effective 'Augment and Criticize' pipeline that mines abundant informative samples for robust detection. To be more specific, in the 'Augment' stage, we present the Augmentation-based Prediction aGgregation (APG), which applies automatically learned transformations to unlabeled images and aggregates detections from various augmented views as pseudo labels. Since not all the pseudo labels from APG are beneficially informative, the subsequent 'Criticize' phase is introduced. Particularly, we present the Critical Retraining Strategy (CRS) that, unlike simply filtering pseudo labels using a fixed threshold, employs a learnable network to evaluate the contribution of unlabeled images at different training timestamps. This way, the noisy samples prohibitive to model evolution can be effectively suppressed. In order to validate 'Augment-Criticize', we apply it to MonoDLE [1] and MonoFlex [2], and the two new detectors, dubbed 3DSeMo$_{\text{DLE}}$ and 3DSeMo$_{\text{FLEX}}$, achieve state-of-the-art results with consistent improvements, evidencing its effectiveness and generality.**

I. Introduction

Monocular 3D (Mono3D) object detection plays an essential role for agents in understanding the real world. However, due to its ill-posed property [3], Mono3D detection remains an open problem. Recently, with the flourishing of deep learning, the community seeks to circumvent the mathematical depth estimation and solve the task in a data-driven manner. A plethora of deep models have been designed and demonstrated remarkable performance [1], [2], [4]–[6]. Despite this, current data volume in Mono3D object detection is nowhere enough for achieving a human-level 3D sensing ability. Since manually annotating 3D boxes in larger-scale data is costly, we argue that semi-supervised learning could be an economical substitute.

Semi-supervised learning, given a *small* amount of manually annotated training samples, aims to explore beneficial information from massive *unlabeled* data for training. It has been extensively studied in 2D vision tasks, such as classification [7]–[10], detection [11]–[13] and segmentation [14]–[16], yet surprisingly less explored for Mono3D detection. One possible reason is that its ill-posed task definition makes algorithms suffer from complex environments and eventually incurs noisy pseudo-label generation on the unlabeled data, degrading performance. A previous method of [17] attempts to construct the multi-view consistency for semi-supervised learning of Mono3D object detection. It is, however, restricted to stereo or multi-view requirement that may be not easily guaranteed in practical scenarios. Instead, in this work, we tend to construct a semi-supervised Mono3D detection framework using only single-camera, -view, and -modality inputs, without relying on the multi-view, or any other modality (*e.g.*, LiDAR) information.

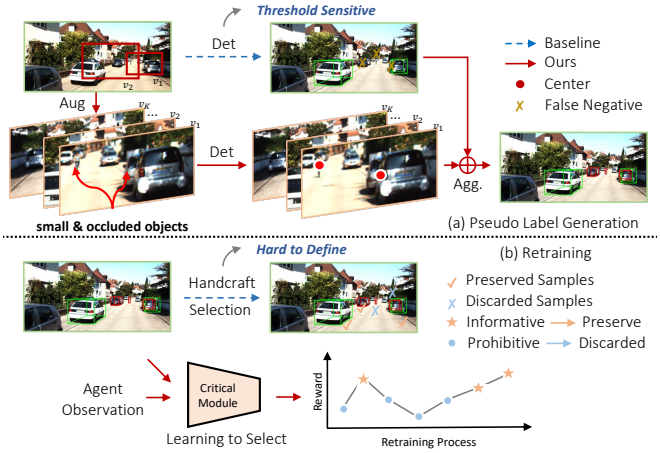

Fig. 1. **Motivation and Proposal.** The differences between our method (red) and previous semi-supervised learning framework (green) in pseudo label (PL) generation and student model retraining. The introduced framework can improve detection recall by observing different views of an image (red dots in (a)), and dynamically determine when to discard an unlabeled sample during training (line-chart in (b)) by the learnable critical module.

Revisiting the road map of semi-supervised learning, we observe that, the bottleneck always lies in the lack of abundant reliable and informative samples from unlabeled data for training. Specifically, two challenges are faced: *How to robustly generate high-quality pseudo labels for unlabeled data* **and** *How to properly leverage these pseudo labels for effective learning*. The former mainly focuses on the quality of training samples, while the latter is related to the evolving direction of the detection model. Yet as discussed earlier, it is *non-trivial* to generate precise pseudo labels under the pure monocular 3D premise. Previous methods in 2D tasks (*e.g.*, [11], [18]) try to alleviate this issue by carefully choosing a threshold for detection box selection (see Fig. 1). This handcraft selection, however, may cause overfit to a specific model, limiting the resilience of the semi-supervised method. Addressing this problem, ***a robust pseudo label generation strategy is thus necessary***. In

*Equal contribution. [1] School of Artificial Intelligence, Shanghai Jiao Tong University. [2] King Abdullah University of Science and Technology. [3] State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS), CASIA. [4] School of Artificial Intelligence, University of Chinese Academy of Sciences. [5] Department of CSE, University of North Texas. Heng Fan is not supported by any funds for this work.

addition, another essential challenge for semi-supervised Mono3D object detection is the lack of an adaptive training strategy to deal with pseudo labels with different qualities. Intuitively the high-quality pseudo labels should contribute more to model updating, and meanwhile the influence of low-quality samples should be suppressed. The classification score (or its variants) of a detection box is adopted in 2D methods (*e.g.*, [11]) to filter out the low-quality samples. But such a handcrafted rule-based manner is hard to guarantee that the model evolves along the right direction, because a sample could show diverse effects to the model at different training steps. Therefore, ***a more adaptive mechanism is needed to guide the training on unlabeled data***.

Contribution. Motivated by the above, we propose a novel 'Augment-Criticize' framework to approach the two challenges in semi-supervised monocular 3D object detection.

In specific in 'Augment' stage, we introduce a simple yet effective Augmentation-based Prediction aGgregation strategy, dubbed **APG**, that aims at robustly generating pseudo labels for unlabeled data (*i.e.*, the first challenge). The core idea is to aggregate predictions from different observations of an image, which we find effectively reduces the detection bias and improves the robustness of pseudo label generation (see Fig. 1 again). In order to avoid handcrafted selection, the transformation parameters for generating an observation are automatically learned by using an efficient reward-based Tree-Parzen-Window algorithm [19]. Interestingly, we find that, content-based transformations such as color-jitter are not helpful, yet geometry-based transformations like resize and crop exhibit much affirmative effect, for improving detection recall, providing guidance for future research.

Since not all the pseudo labels from APG is beneficially informative, an adaptive strategy is desired to exploit these pseudo labels for effective model training (*i.e.*, the second challenge), which motivates the proposed 'Criticize' stage. More specifically, in this stage, a Critical Retraining Strategy (**CRS**) is imposed to adaptively update the model with noisy pseudo labels. Particularly, CRS contains a memory bank to preserve evaluation images and a critical module to determine which pseudo label benefits to update the model. At each training step, the loss of each pseudo label corresponds to an update choice of the model. The critical module samples images from the memory to determine whether this update improves model capability. If the model resembles to a worse one, the update would be discarded (self-criticise). During the cyclical updating of the memory bank, the critical module gradually encodes the knowledge of the whole evaluation set to its weight parameters, and therefore it becomes more powerful along training period.

To verify our 'Augment-Criticize' framework, we apply it to MonoDLE [1] and MonoFlex [2]. In experiments, compared with baselines, our semi-supervised detectors, 3DSeMo$_{\text{DLE}}$ and 3DSeMo$_{\text{FLEX}}$, achieve consistent improvements for about 3% $\texttt{CAR}(\texttt{Mod.})$ $\texttt{AP}_{\texttt{3D}}$ on KITTI, which shows the effectiveness and versatility of our method.

In summary, we make the following contributions:

(**1**) *We propose a novel 'Augment-Criticize' framework for semi-supervised Mono3D object detection.*
(**2**) *We propose an augmented-based prediction aggregation to improve the quality of pseudo labels for unlabeled data.*
(**3**) *We propose a critical retraining strategy that adaptively evaluates each pseudo label for effective model training.*
(**4**) *We integrate our semi-supervised framework into different methods, and results evidence its effectiveness.*

II. Related Work

A. Monocular 3D Object Detection

Mono3D object detection, which only requires vision clues from a single camera, is a widely applied solution for agents to perceive the 3D world [1], [2], [20]–[27]. Earlier attempts devoted massive efforts to the ill-posed depth estimation problem by adopting an isolated depth model [28] to generate pseudo point cloud or lifting 2D features to 3D space [29]. Despite the promising results, the hefty computation overhead entailed by dense depth estimation prohibits such methods from practical applications. Later the depth estimation is moved to an auxiliary head, [1], [2], [4], [6], which enables end-to-end model training with a neater framework. Representative methods like SMOKE [4] and MonoDLE [1] adopt CenterNet-like architectures [30], whereas FCOS3D [6] and PGD [31] extend the 2D FCOS detector [32] into a 3D detection model. In this paper, we aim to design a general semi-supervised framework, which is agnostic of and robust to specific model designs, to push the evolution of modern Mono3D object detectors.

B. Semi-Supervised Learning

Semi-Supervised Learning (SSL) is attractive because of its capability to further unveil the power of machine learning with abundant cheap unlabeled data [8], [9], [33]–[39]. Due to the space limitation, this section only reviews self-training-based methods, which is one of the most engaging directions that has been studied for decades [40], [41]. In general, self-training-based semi-supervised learning methods first train a teacher model with a small set of human-annotated data. The teacher model then generates pseudo labels on a much larger set of unlabeled data. Finally, a student model is trained with both human-labeled and self-annotated data. Such a paradigm has demonstrated great success in image classification [7]–[10], semantic segmentation [14]–[16], and 2D object detection [11]–[13], [42]. While different applications usually require additional bells and whistles, the core components of semi-supervised learning remain unchanged: how to generate high-quality pseudo-label, and how to effectively utilize the pseudo-label to train student models. Mean-Teacher [7] proposes temporal ensembling to facilitate retraining. Soft-Teacher [11] utilizes the classification score to reweight supervision on the student model and imposed 2D bounding box jitter to filter unreliable pseudo labels. ST++ [14] adopts strong augmentations on the unlabeled samples and leverages evolving stability during training to prioritize high-quality labels. Compared with well-studied 2D tasks, it is much more challenging for Mono3D object

detection to collect reliable pseudo labels. Although such issue can be alleviated by introducing multi-view consistency [17], compared with abundant single-view datasets, high-quality stereo or multi-view datasets are much harder to collect. Besides, learning consistency among video frames is vulnerable to moving objects.

III. Method

A. Preliminary

Task Definition. Given an image sample x in the labeled dataset, its label y contains information about the category, location, dimension, and orientation of objects visible in x. Semi-supervised Mono3D object detection aims to acquire knowledge from both annotated dataset $\mathcal{D}_l = \{x_i^l, y_i^l\}_{i=1}^{N_l}$ and unlabeled dataset $\mathcal{D}_u = \{x_j^u\}_{j=1}^{N_u}$, where $N_u \gg N_l$.

Vanilla Self-Training Scheme. Self-training is a prominent branch in semi-supervised learning [14], [43]. A vanilla self-training [14] pipeline contains three major steps: 1) Standard Supervised Training: which trains a teacher model M_t on the labeled dataset $\mathcal{D}_l$, 2) Pseudo Label Generation: which predicts pseudo labels $\{\hat{y} = M_t(x_j) | x_j \in \mathcal{D}_u\}$ on the unlabeled dataset $\mathcal{D}_u$, and 3) Retraining with Noisy Labels: which learns a student model M_s for final evaluation. Using M_s as the new teacher, the step 2 and 3 can be repeated until satisfactory performance is obtained.

In this paper, we elaborately investigate the pseudo label generation (step 2) and retraining strategy (step 3), which are the most crucial parts of the self-training scheme. To fully demonstrate the simplicity of the proposed semi-supervised framework, we don't iteratively perform step 2 and 3.

B. Augmentation-Based Prediction Aggregation

To obtain high-quality pseudo labels of the unlabeled data, previous 2D semi-supervised learning methods [6], [11], [12], [18], [44] resort to a suitable threshold τ to filter predicted boxes. However, it is non-trivial to determine an optimal threshold for each different method. In order to alleviate the dependency on such a handcrafted threshold, we propose the APG to improve the robustness of pseudo-label generation by effectively aggregating predictions from different observations of the same image. The proposed algorithm, that is illustrated in Alg. 1, consists of three steps:

1) Firstly, given an input image from the unlabeled dataset $x_j^u \in \mathcal{D}_u$, the teacher model M_t predicts the detection results for x_j^u and its K augmented images. Let $\mathcal{P}_r$ denote the raw prediction of x_j^u, and $\mathcal{P}_f^0$ and $\{\mathcal{P}_f^k\}_{k=1}^K$ represent the post-processed (by the pre-defined threshold τ) predictions of x_j^u and the augmented images, respectively.

2) Secondly, for each predicted box p_i in $\mathcal{P}_f^0$, we apply the kNN clustering algorithm to find its nearest neighbors in $\{\mathcal{P}_f^k\}_{k=1}^K$, that forms a cluster. p_i is considered as a pseudo label for x_j^u. Intuitively, the number of assigned predictions n in the cluster indicates the difficulty degree in detecting an object, and the variance σ by Maximum Likelihood Estimation (MLE) measures the uncertainty of p_i. With the classification score s, these by-products are combined by Eq. 1 to demonstrate p_i's reliability, which is then used to weight the loss of each unlabeled data in retraining.

$$w = \gamma_1 \times s + (1 - \gamma_1) \times \exp\left(-\frac{\sigma}{n} * \gamma_2\right), \tag{1}$$

We set $\gamma_1 = 0.6$ and $\gamma_2 = 6$ in our model, respectively.

3) Finally, for the unused predictions in $\{\mathcal{P}_f^k\}_{k=1}^K$, they would be self-clustered. The cluster centers are treated as reference points, whose closest prediction in $\mathcal{P}_r$ are selected as pseudo labels. Their uncertainties are measured by Eq. 1.

Moreover, inspired by successful attempts at auto data augmentations [45], [46], we resort to the Tree-Structured Parzen Estimators (TPE) [19] to automatically pick the K transformations and their hyper-parameters (*e.g.*, resize ratio). More details are presented in supplementary materials.

Algorithm 1: APG-u Pseudocode

Input: Predictions of different observations,
Raw prediction $\mathcal{P}_r$ of an unlabeled image,
Filtered prediction $\mathcal{P}_f^0$, $N = \text{len}(\mathcal{P}_f^0)$,
Filtered predictions $\{\mathcal{P}_f^k\}_{k=1}^K$ of augmented images,
Threshold τ for kNN

Output: Aggregated prediction $\mathcal{P}$

Initialize set $\mathcal{S} \leftarrow \{\{p_1\}, \cdots, \{p_N\}\}, p_n \in \mathcal{P}_f^0$
for *image observation* $k \in \{1, \ldots, K\}$ **do**
  for *prediction* $p_i \in \mathcal{P}_f^k$ **do**
    (index j, distance l) $\leftarrow$ kNN$(p_i, \mathcal{S})$
    if $l < \tau$ **then**
      $\mathcal{S}_j \leftarrow \mathcal{S}_j \cup \{p_i\}$ # append p_i to clusters
    else
      $\mathcal{S}_j \leftarrow \mathcal{S}_j \cup \{\{p_i\}\}$ # create new clusters
for *loop index n, cluster set* $\{p^m\}_{m=1}^M \in \mathcal{S}$ **do**
  location μ, variance σ = MLE $(\{p^m\}_{j=m}^M)$
  if $n < N$ **then**
    $\mathcal{P} \leftarrow \mathcal{P} \cup \{(\mathcal{S}_n[0], \sigma)\}$
  else
    $\mathcal{P} \leftarrow \mathcal{P} \cup \{(\text{NearestSearch}(\mu, \mathcal{P}_r), \sigma)\}$
return $\mathcal{P}$

C. Critical Retraining Strategy

The generated pseudo labels inevitably contain noises, thus it is crucial to explore informative ones that benefit model evolution. The uncertainty measurements of pseudo labels provided by the APG module can enhance the stability of retraining, but it still suffers from the fixed weight of each sample. We argue that the contribution of each sample during model training should adapt to the model's state as training proceeds. To this end, we propose a learning-based critical module to adaptively find the informative unlabeled data, which may provide a new perspective for semi-supervised Mono3D object detection to mine informative samples.

Specifically, the critical module first evaluates the effect of a training sample from the unlabeled dataset, and then assigns it with a 0-1 binary flag indicating whether to back-propagate its gradients. From a reinforcement learning perspective, we regard the student model as an *agent*, its weight parameters as the *state*, the input image and the output of the model as an *observation*. At state $\mathcal{S}$, a detection

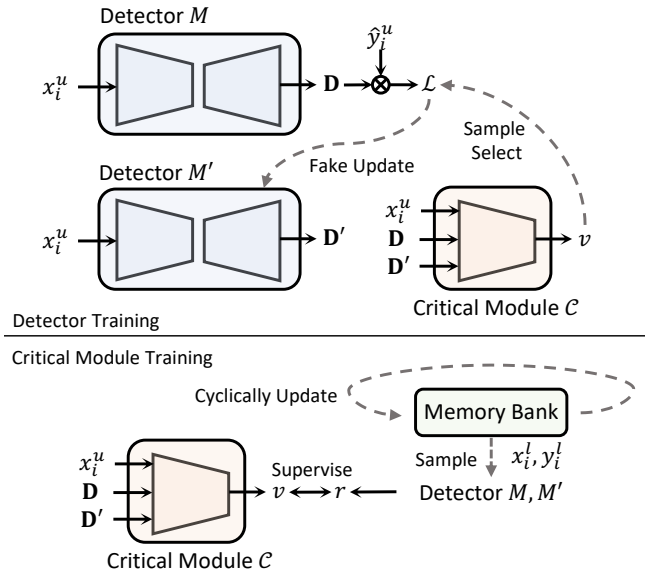

Fig. 2. **Illustration of CRS.** We adopt a critical module to determine whether a sample from the unlabeled data benefits model convergence. The memory bank is cyclically updated.

loss $\mathcal{L}_{unsup}$ for the agent can be calculated based on the given observation $\mathcal{O}$. If the gradients of $\mathcal{L}_{unsup}$ are back-propagated, the state will be updated to $\mathcal{S}^{'}$ and the model output (observation) will be updated to $\mathcal{O}^{'}$. Our proposed critical module then evaluates whether $\mathcal{S}^{'}$ is the optimal choice of updated $\mathcal{S}$ based on the observations $\mathcal{O}$ and $\mathcal{O}^{'}$.

At each training step, an input image x_i^u from the unlabeled data is fed into the detector M (agent), obtaining the predictions $\mathbf{D}$ (classification and regression response maps),

$$\mathbf{D} = M(x_i^u|\mathcal{S}), \tag{2}$$

With the pseudo label $\hat{y}_i^u$, we can get the training loss,

$$\mathcal{L}_{unsup} = \mathcal{L}(\mathbf{D}, \hat{y}_i^u), \tag{3}$$

We take one '*trial*' gradient descent step to obtain the updated model M' with parameters $\mathcal{S}^{'}$. Then, the critical module evaluates the effectiveness of this update ($\mathcal{S} \rightarrow \mathcal{S}^{'}$),

$$v = \mathcal{C}(x_i^u, \mathbf{D}, \mathbf{D}^{'}|\Psi), \tag{4}$$

where $\mathbf{D}^{'}$ is the predictions of the updated model M' on x_i^u, and Ψ is the parameter of the critical module. When v is larger than 1, x_i^u will be considered as an informative sample, and this update is retained. Otherwise, the update is discarded, and the model parameters will be reverted to S. The scheme is illustrated in Fig. 2.

Moreover, it is crucial to design a feasible training objective to make sure the cirtical module can provide reliable feedback. In our work, we propose a reward function to supervise the training of the critical network,

$$r = \mathcal{L}(M(x_i|\mathcal{S}), y_i) - \mathcal{L}(M'(x_i|\mathcal{S}^{'}), y_i), \tag{5}$$

where (x_i, y_i) denotes samples from the training set of the labeled dataset $\mathcal{D}_l$. The L2 loss is applied to v and r for supervising the learning of critical module. During training, we alternately update the detector and critical module.

Notably, it's impractical to evaluate all samples to get a reliable reward r due to the unaffordable computation cost. Motivated by the self-supervised method MoCo [47], we employ a memory bank (queue) to buffer the training samples in $\mathcal{D}_l$ and cyclically update it.

IV. Experiments

A. Experimental Setup

Dataset. We conduct experiments on both KITTI [57] and Waymo [58]. KITTI is commonly used for evaluation in semi-supervised monocular 3D object detection methods. But few works evaluate the large-scale Waymo benchmark. In our work, we conduct experiments on both of them to show the effectiveness and generality of our model. For KITTI, following [59], we split the original training set into 3,712 training and 3,769 validation samples. The unlabeled data contains 33,507 samples obtained from the official unlabeled videos in KITTI. Similarly in Waymo, 3162 samples (only about 2%) from the front cameras are used as annotated samples, and the left 154919 samples are used as unlabeled ones. The metrics to indicate performance follow the official design of each benchmark. We will provide detailed image information upon code release.

Implementation Details. We integrate the proposed semi-supervised framework to classical Mono3D detectors MonoDLE [1] and MonoFlex [2]. Unless otherwise specified, the proposed APG augments an input image from the unlabeled dataset to $K = 9$ different views. While the initial threshold for filtering detection boxes is set as 0.65, other predictions with confidence scores lower than 0.65 will be used in the center aggregation algorithm. For the proposed CRS, we construct the critical module with ResNet-50 [60]. Notably, the critical module is not used during inference. For a batch size of 8, we chop off the 2 samples with lowest evaluation value v in CRS training. For fair comparisons, we reproduce the baseline methods MonoDLE and MonoFlex based on the official codes provided by the authors. While most Mono3D methods are trained on a single GPU, we adopt 8 A6000 GPUs in all experiments to facilitate training with a larger data volume. Ablations are conducted based on MonoDLE and evaluated on KITTI unless otherwise specified. For simplicity, the proposed copy-paste is not used in ablation experiments unless otherwise specified.

B. Main Results

Quantitative comparisons of our method with other state-of-the-art models on the KITTI leaderboard are presented in Tab. I and Tab. II. Also notably, few semi-supervised Mono3D methods provide performance on challenging categories of pedestrian and cyclist on KITTI. It shows that by effectively leveraging larger volumes of unlabeled data, our proposed semi-supervised strategy significantly boosts the performance of the baseline methods. In particular, our approach improves the baseline MonoDLE by +3.32%/+2.89% on $\mathtt{AP}_{\mathtt{3D}}(\mathtt{Mod.})$ and $\mathtt{AP}_{\mathtt{BEV}}(\mathtt{Mod.})$ of the KITTI test set, respectively. The gains on $\mathtt{AP}(\mathtt{Easy})$ of our $\text{3DSeMo}_{\text{DLE}}$ surprisingly exceeds +5% on all metrics and data splits of KITTI.

TABLE I

COMPARISION WITH STATE-OF-THE-ART (SOTA) METHODS ON KITTI CAR. WE PRESENT THE EVALUATION RESULTS OF 'CAR' IN THE KITTI TEST AND VALIDATION SETS. METHODS ARE SORTED BASED ON THE RESULTS OF CAR(Mod.) AP_{3D} ON TEST SET.

Method		Test AP_{3D}			Test AP_{BEV}			Val AP_{3D}		
		Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
PatchNet [48]	Depth	15.68	11.12	10.17	22.97	16.86	14.97	-	-	-
D4LCN [22]		16.65	11.72	9.51	22.51	16.02	12.55	-	-	-
DDMP-3D [49]		19.71	12.78	9.80	28.08	17.89	13.44	-	-	-
Kinematic3D [21]	Multi-frames	19.07	12.72	9.17	26.69	17.52	13.10	19.76	14.10	10.47
MonoRUn [50]	LiDAR	19.65	12.30	10.58	27.94	17.34	15.24	20.02	14.65	12.61
CaDDN [51]		19.17	13.41	11.46	27.94	18.91	17.19	23.57	16.31	13.84
MonoDTR [25]		21.99	15.39	12.73	28.59	20.38	17.14	24.52	18.57	15.51
AutoShape [52]	CAD	22.47	14.17	11.36	30.66	20.08	15.59	20.09	14.65	12.07
SMOKE [4]	None	14.03	9.76	7.84	20.83	14.49	12.75	14.76	12.85	11.50
MonoPair [53]		13.04	9.99	8.65	19.28	14.83	12.89	16.28	12.30	10.42
RTM3D [54]		13.61	10.09	8.18	-	-	-	19.47	16.29	15.57
PGD [31]		19.05	11.76	9.39	26.89	16.51	13.49	19.27	13.23	10.65
MonoRCNN [23]		18.36	12.65	10.03	25.48	18.11	14.10	16.61	13.19	10.65
Zhang *et al.*$_{\text{DLE}}$ [55]		20.25	14.14	12.42	28.85	17.72	17.81	20.82	15.64	13.82
GUPNet [24]		20.11	14.20	11.77	-	-	-	22.76	16.46	13.72
HomoLoss$_{\text{FLEX}}$ [56]		21.75	14.94	13.07	29.60	20.68	17.81	23.04	16.89	14.90
MonoDLE [1]	None	17.23	12.26	10.29	24.79	18.89	16.00	17.45	13.66	11.68
3DSeMo$_{\text{DLE}}$	Unlabeled	**23.11**	**15.58**	**13.58**	**30.99**	**21.78**	**18.64**	**22.87**	**17.65**	**14.83**
Improvement	*v.s. baseline*	+5.88	+3.32	+3.29	+6.20	+2.89	+2.64	+5.42	+3.99	+3.15
MonoFlex [2]†	None	19.94	13.89	12.07	28.23	19.75	16.89	21.62	16.05	13.40
3DSeMo$_{\text{FLEX}}$	Unlabeled	**23.55**	**15.25**	**13.24**	**32.57**	**21.21**	**18.07**	**25.14**	**18.65**	**15.58**
Improvement	*v.s. baseline*	+3.61	+1.36	+1.17	+4.34	+1.46	+1.18	+3.52	+2.60	+2.18

TABLE II

COMPARISONS ON 'PEDESTRIAN' AND 'CYCLIST' OF KITTI.

Method	Ped. AP_{3D} IoU $\geqslant 0.5$			Cyc. AP_{3D} IoU $\geqslant 0.5$		
	Easy	Mod.	Hard	Easy	Mod.	Hard
baseline	9.64	6.55	5.44	4.59	2.66	2.45
3DSeMo$_{\text{DLE}}$	**10.78**	**7.26**	**6.05**	**7.04**	**4.24**	**3.56**
Improvement	+1.14	+0.71	+0.61	+2.45	+1.58	+1.11

TABLE III

COMPARISONS ON WAYMO VALIDATION SET.

Method	Veh.		Ped.		Cyc.	
	mAP	mAPL	mAP	mAPL	mAP	mAPL
baseline	41.53	24.91	13.34	7.87	5.13	3.17
+10%Unlabel	**49.13**	**30.26**	**16.52**	**9.87**	**8.63**	**5.22**
Improvement	+7.60	+5.35	+3.18	+2.00	+3.50	+2.05
+100%Unlabel	**53.11**	**34.37**	**19.41**	**11.83**	**12.89**	**7.73**
Improvement	+11.58	+9.46	+6.07	+3.96	+7.76	+4.56

When integrating our method to MonoFlex [61], it achieves gains of +3.61/1.36 on AP_{3D}(Easy/Mod.), respectively, evidencing the generality of our framework. Moreover, on the challenging categories of KITTI (see Tab. II) and the much larger benchmark Waymo (see Tab. III), our method consistently demonstrates performance improvements, providing clear validation of the efficacy and applicability of the proposed approach. Remarkably, even with a mere 2% labeled data and 10% unlabeled data of Waymo, our method achieves a substantial 7.6% improvement in mAP for Vehicle. When leveraging all unlabeled data, the mAP for Vehicle shows an impressive gain of 11.58%. The consistent advancements across challenging categories such as pedestrian and cyclist underscore the efficacy of our proposed method.

C. Ablation Studies and Analysis

1) Overall Component-wise Analysis: To understand the effect of each component, we incrementally apply the proposed APG and CRS to the baseline detector MonoDLE [1] with Car AP_{3D}(Mod.) of 13.66. As shown in Tab. IV, the vanilla self-training strategy improves the baseline model for 2.11% on Car AP_{3D}(Mod.) without bells and whistles. Subsequently, we apply the proposed APG and CRS to the model, respectively. It shows that both of them can significantly improve Car AP_{3D}(Mod.) by about 2.5%, which validates our argument that robust pseudo label generation and finding informative samples are both crucial for semi-supervised Mono3D object detection. Last but not least, while maintaining the superiority on Car, combining the proposed APG and CRS can pre-eminently improve Ped.(Mod.) AP_{3D} for about 1.4%. In Mono3D object detection, the pedestrian is more challenging than car because of the much smaller object size, for which a slight prediction shift leads to drastic degradation of IoU. The pseudo labels of pedestrian thus contain much more noise. Therefore, the gains on pedestrian prove CRS's ability in adaptively selecting informative samples from severely noisy pseudo labels.

TABLE IV

COMPONENT-WISE ANALYSIS.

Method	Car AP_{3D} IoU $\geqslant 0.7$			Ped. AP_{3D} IoU $\geqslant 0.5$		
	Easy	Mod.	Hard	Easy	Mod.	Hard
Baseline	16.77	13.66	11.54	5.53	4.45	3.33
Plainest Self-Training	20.14	15.77	13.27	7.27	5.99	4.74
+ CRS	22.64	17.53	14.59	9.43	6.81	5.71
+ APG	22.71	17.56	14.68	9.35	6.77	5.58
+ APG + CRS	**22.87**	**17.65**	**14.83**	**10.99**	**8.25**	**6.72**

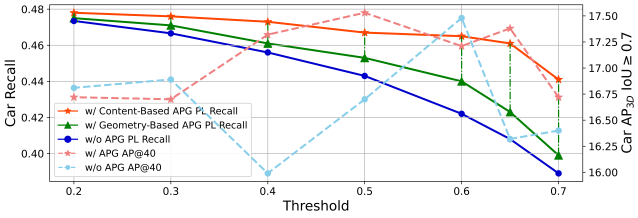

Fig. 3. **Effectiveness of APG.** We demonstrate the robustness on both pseudo label quality (recall) and 3D object detection performance (AP@40).

2) Robustness of APG: Previous semi-supervised methods usually filter detection boxes to generate pseudo labels by applying a threshold τ on the classification score. However,

TABLE V

EFFECTIVENESS OF REWEIGHTING STRATEGY.

	Method	Car AP_{3D} IoU $\geqslant 0.7$			Ped. AP_{3D} IoU $\geqslant 0.5$		
		Easy	Mod.	Hard	Easy	Mod.	Hard
①	w/o APG	21.75	16.32	14.15	8.34	6.04	4.80
②	w/ APG	22.66	17.38	14.67	7.71	5.88	4.74
③	② + reweight	**22.71**	**17.56**	**14.68**	**9.35**	**6.77**	**5.58**

TABLE VI

EFFECTIVENESS OF CRS. 'CM.' DENOTES THE PROPOSED CRITICAL MODULE.

	Method	Car AP_{3D} IoU $\geqslant 0.7$			Ped. AP_{3D} IoU $\geqslant 0.5$		
		Easy	Mod.	Hard	Easy	Mod.	Hard
①	Baseline	22.71	17.56	14.68	9.35	6.77	5.58
②	bbox jitter filter	22.50	16.71	14.40	8.36	6.30	5.07
③	score filter	21.86	17.20	14.31	9.04	7.25	5.75
④	CRS w/o cm.	22.01	16.18	13.95	7.05	5.58	4.26
⑤	CRS w/ cm.	**22.87**	**17.65**	**14.83**	**10.99**	**8.25**	**6.72**

as presented in Fig. 3 (the blue solid line), it suffers from a drastic degradation on `Recall` when enlarging the threshold. Besides, its detection performance is sensitive to threshold change (the cyan dotted line). In contrast, the performances of our APG (the red dotted lines) are more stable, which proves its robustness in generating pseudo labels. We select $\tau = 0.65$ in our model based on the observation of this experiment, with which the APG can boost the `Car` (`Mod.`) AP_{3D} for 1.06%, as shown in Tab. V. Though we need to set an initial threshold in APG, our experiment (the red solid line) shows that it is less sensitive to threshold change. Fig. 3 also shows that geometry-based augmentation (*e.g.* resize, the red solid line) is superior to the content-based counterpart (*e.g.* color jitter, the green solid line) in improving the quality of the generated pseudo labels. This may attribute to their different mechanism that content-based transformations only marginally modify the context, while geometry-based transformations can significantly migrate the position and scale distribution of objects, which are the common reasons for false negatives in Mono3D object detection. Notably, the transformations are automatically learned by TPE, which can be effortlessly integrated into other detectors. All details and source codes as well as the results about TPE will be released for reproduction purpose.

3) Sample Weight from APG: While APG improves the overall recall of pseudo labels, it inevitably introduces more noise to challenging categories (*e.g.,* pedestrian) as shown in Tab. V. As a result, the `Ped.` (`Mod.`) AP_{3D} drops 0.16% (② *v.s.* ①). To alleviate this, we weight the loss of each unlabeled sample in the retraining phase with the by-product clues from the proposed APG (see Eq. 1). As shown in Tab. V, our strategy can not only avoid performance degradation but also impressively obtain 0.89% improvement on the `Ped.` (`Mod.`) AP_{3D} (③ *v.s.* ②), which shows the versatility of the proposed APG.

4) Different strategies for selecting informative samples: The proposed CRS aims to adaptively separate informative samples from noisy ones. To demonstrate the superiority of CRS, we compare against some alternative strategies which have demonstrated success in other tasks. The compared counterparts include 1) filtering samples with the quality score of pseudo labels introduced in Eq. 1, 2) the bbox jitter proposed for 2D detection in [11]. We tailor the 2D box jitter strategy for 3D detection, and details are presented in the supplementary material. As shown in Tab. VI, bbox jitter causes performance degradation because of its unreliable quality measurement for pseudo labels (② *v.s.* ①). ③ throws away unreality samples based on classification and location scores (see Eq. 1). It shows that ③ only slightly improves pedestrian detection performance, however still lagging behind our proposed CRS (⑤). Besides the unreliability of detection scores and box jitter in Mono3D, another underlying reason for the advance of CRS is that ① and ② are static strategies where the filtering indicator of a sample holds along the retraining phase. Conversely, the indicator learned by the critical module changes in different retraining timestamps, as shown in Fig. 4. It both intuitively and theoretically makes sense that the importance of a sample should be mutative in training.

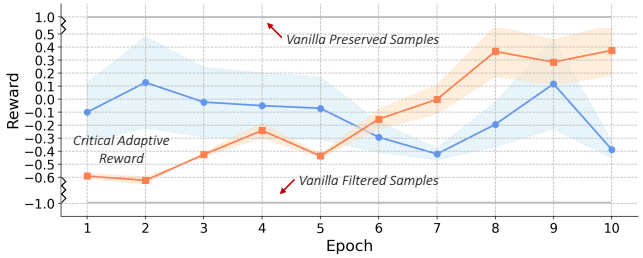

Fig. 4. **Adaptive Reward of CRS.** Vanilla strategies invariably drop out or preserve a sample whereas our critical module predicts adaptive rewards.

Learnable or not. The proposed CRS learns the filtering indicator with a learnable critical module (Eq. 4). Yet intuitively, we can simply determine the contribution of a sample by the training loss before and after the model updating with Eq. 5. To validate the necessity of the proposed scheme, we prohibit the critical module and directly leverage the reward calculated in Eq. 5 as the indicator to select samples during retraining. As shown in Tab. VI ④, unsurprisingly, this naive strategy degrades the overall performance because of its biased optimization objective. In particular, the strategy of ④ can only access several samples during calculating the reward, lacking the global vision of the evaluation set. In contrast, the learning-based critical module encodes the knowledge of the whole dataset to its weights parameters through cyclically updating the memory bank, which can provide better indicators for model retraining (see ⑤).

V. CONCLUSION

In this paper, we propose the novel 'Augment and Criticize' policies to construct a general framework for self-training-based semi-supervised monocular 3D object detection by exploring informative samples. The proposed APG is able to aggregate samples from different views of unlabeled images and copy-paste augmentation for robust label generation. On the other hand, the CRS adopts a learnable critical module to measure the reward of each pseudo sample and filter noisy ones to enhance model training. Our extensive experiments and analyses on multiple benchmarks demonstrate the effectiveness of our approach.

REFERENCES

[1] X. Ma, Y. Zhang, D. Xu, D. Zhou, S. Yi, H. Li, and W. Ouyang, "Delving into localization errors for monocular 3d object detection," in *CVPR*, 2021, pp. 4721–4730.

[2] Y. Zhang, J. Lu, and J. Zhou, "Objects are different: Flexible monocular 3d object detection," in *CVPR*, 2021, pp. 3289–3298.

[3] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *NeurIPS*, vol. 27, 2014.

[4] Z. Liu, Z. Wu, and R. Tóth, "Smoke: Single-stage monocular 3d object detection via keypoint estimation," in *CVPRW*, 2020, pp. 996–997.

[5] R. Zhang, H. Qiu, T. Wang, X. Xu, Z. Guo, Y. Qiao, P. Gao, and H. Li, "Monodetr: Depth-aware transformer for monocular 3d object detection," *arXiv preprint arXiv:2203.13310*, 2022.

[6] T. Wang, X. Zhu, J. Pang, and D. Lin, "Fcos3d: Fully convolutional one-stage monocular 3d object detection," in *ICCV*, 2021, pp. 913–922.

[7] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," *NeurIPS*, vol. 30, 2017.

[8] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," *NeurIPS*, vol. 32, 2019.

[9] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," in *NeurIPS*, 2020, pp. 596–608.

[10] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, "Self-training with noisy student improves imagenet classification," in *CVPR*, 2020, pp. 10 687–10 698.

[11] M. Xu, Z. Zhang, H. Hu, J. Wang, L. Wang, F. Wei, X. Bai, and Z. Liu, "End-to-end semi-supervised object detection with soft teacher," in *ICCV*, 2021, pp. 3060–3069.

[12] H. Zhou, Z. Ge, S. Liu, W. Mao, Z. Li, H. Yu, and J. Sun, "Dense teacher: Dense pseudo-labels for semi-supervised object detection," *arXiv preprint arXiv:2207.02541*, 2022.

[13] F. Zhang, T. Pan, and B. Wang, "Semi-supervised object detection with adaptive class-rebalancing self-training," in *AAAI*, 2022, pp. 3252–3261.

[14] L. Yang, W. Zhuo, L. Qi, Y. Shi, and Y. Gao, "St++: Make self-training work better for semi-supervised semantic segmentation," in *CVPR*, 2022, pp. 4268–4277.

[15] Y. Liu, Y. Tian, Y. Chen, F. Liu, V. Belagiannis, and G. Carneiro, "Perturbed and strict mean teachers for semi-supervised semantic segmentation," in *CVPR*, 2022, pp. 4258–4267.

[16] J. Yuan, J. Ge, Q. Qian, Z. Wang, F. Wang, and Y. Liu, "Semi-supervised semantic segmentation with mutual knowledge distillation," *arXiv preprint arXiv:2208.11499*, 2022.

[17] Q. Lian, Y. Xu, W. Yao, Y. Chen, and T. Zhang, "Semi-supervised monocular 3d object detection by multi-view consistency," in *ECCV*, 2022.

[18] B. Chen, W. Chen, S. Yang, Y. Xuan, J. Song, D. Xie, S. Pu, M. Song, and Y. Zhuang, "Label matching semi-supervised object detection," in *CVPR*, 2022, pp. 14 381–14 390.

[19] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," *NeurIPS*, vol. 24, 2011.

[20] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3d object detection methods for autonomous driving applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3782–3795, 2019.

[21] G. Brazil, G. Pons-Moll, X. Liu, and B. Schiele, "Kinematic 3d object detection in monocular video," in *ECCV*. Springer, 2020, pp. 135–152.

[22] M. Ding, Y. Huo, H. Yi, Z. Wang, J. Shi, Z. Lu, and P. Luo, "Learning depth-guided convolutions for monocular 3d object detection," in *CVPRW*, 2020, pp. 1000–1001.

[23] X. Shi, Q. Ye, X. Chen, C. Chen, Z. Chen, and T.-K. Kim, "Geometry-based distance decomposition for monocular 3d object detection," in *ICCV*, 2021, pp. 15 172–15 181.

[24] Y. Lu, X. Ma, L. Yang, T. Zhang, Y. Liu, Q. Chu, J. Yan, and W. Ouyang, "Geometry uncertainty projection network for monocular 3d object detection," in *ICCV*, 2021, pp. 3111–3121.

[25] K.-C. Huang, T.-H. Wu, H.-T. Su, and W. H. Hsu, "Monodtr: Monocular 3d object detection with depth-aware transformer," in *CVPR*, 2022, pp. 4012–4021.

[26] H. Sheng, S. Cai, N. Zhao, B. Deng, M.-J. Zhao, and G. H. Lee, "Pdr: Progressive depth regularization for monocular 3d object detection," *IEEE TCSVT*, 2023.

[27] C. Tao, J. Cao, C. Wang, Z. Zhang, and Z. Gao, "Pseudo-mono for monocular 3d object detection in autonomous driving," *IEEE TCSVT*, 2023.

[28] X. Weng and K. Kitani, "Monocular 3d object detection with pseudo-lidar point cloud," in *ICCVW*, 2019, pp. 0–0.

[29] T. Roddick, A. Kendall, and R. Cipolla, "Orthographic feature transform for monocular 3d object detection," *arXiv preprint arXiv:1811.08188*, 2018.

[30] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," *arXiv preprint arXiv:1904.07850*, 2019.

[31] T. Wang, Z. Xinge, J. Pang, and D. Lin, "Probabilistic and geometric depth: Detecting objects in perspective," in *CoRL*. PMLR, 2022, pp. 1475–1485.

[32] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," in *ICCV*, 2019, pp. 9627–9636.

[33] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.

[34] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[35] D.-H. Lee *et al.*, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *ICMLW*, 2013, p. 896.

[36] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko, "Semi-supervised learning with ladder networks," *Advances in neural information processing systems*, vol. 28, 2015.

[37] D. Li, Y. Liu, and L. Song, "Adaptive weighted losses with distribution approximation for efficient consistency-based semi-supervised learning," *IEEE TCSVT*, vol. 32, no. 11, pp. 7832–7842, 2022.

[38] Y. Li, S. Kan, C. Wenming, and Z. He, "Learned model composition with critical sample look-ahead for semi-supervised learning on small sets of labeled samples," *IEEE TCSVT*, vol. 31, no. 9, pp. 3444–3455, 2020.

[39] L. Yang, X. Zhang, J. Li, L. Wang, M. Zhu, C. Zhang, and H. Liu, "Mix-teaching: A simple, unified and effective semi-supervised learning framework for monocular 3d object detection," *IEEE TCSVT*, 2023.

[40] G. J. McLachlan, "Iterative reclassification procedure for constructing an asymptotically optimal rule of allocation in discriminant analysis," *Journal of the American Statistical Association*, vol. 70, no. 350, pp. 365–369, 1975.

[41] H. Scudder, "Probability of error of some adaptive pattern-recognition machines," *IEEE Transactions on Information Theory*, vol. 11, no. 3, pp. 363–371, 1965.

[42] Y. Tang, Z. Cao, Y. Yang, J. Liu, and J. Yu, "Semi-supervised few-shot object detection via adaptive pseudo labeling," *IEEE TCSVT*, 2023.

[43] M.-R. Amini, V. Feofanov, L. Pauletto, E. Devijver, and Y. Maximov, "Self-training: A survey," *arXiv preprint arXiv:2202.12040*, 2022.

[44] Z. Li, Z. Chen, A. Li, L. Fang, Q. Jiang, X. Liu, and J. Jiang, "Unsupervised domain adaptation for monocular 3d object detection via self-training," *arXiv preprint arXiv:2204.11590*, 2022.

[45] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation policies from data," *arXiv preprint arXiv:1805.09501*, 2018.

[46] S. Lim, I. Kim, T. Kim, C. Kim, and S. Kim, "Fast autoaugment," *NeurIPS*, vol. 32, 2019.

[47] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.

[48] X. Ma, S. Liu, Z. Xia, H. Zhang, X. Zeng, and W. Ouyang, "Rethinking pseudo-lidar representation," in *ECCV*, 2020.

[49] L. Wang, L. Du, X. Ye, Y. Fu, G. Guo, X. Xue, J. Feng, and L. Zhang, "Depth-conditioned dynamic message propagation for monocular 3d object detection," in *CVPR*, 2021, pp. 454–463.

[50] H. Chen, Y. Huang, W. Tian, Z. Gao, and L. Xiong, "Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation," in *CVPR*, 2021, pp. 10 379–10 388.

[51] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander, "Categorical depth distribution network for monocular 3d object detection," in *CVPR*, 2021, pp. 8555–8564.

[52] Z. Liu, D. Zhou, F. Lu, J. Fang, and L. Zhang, "Autoshape: Real-time shape-aware monocular 3d object detection," in *ICCV*, 2021, pp. 15 641–15 650.
[53] Y. Chen, L. Tai, K. Sun, and M. Li, "Monopair: Monocular 3d object detection using pairwise spatial relationships," in *CVPR*, 2020, pp. 12 093–12 102.
[54] P. Li, H. Zhao, P. Liu, and F. Cao, "Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving," in *ECCV*. Springer, 2020, pp. 644–660.
[55] Y. Zhang, W. Zheng, Z. Zhu, G. Huang, D. Du, J. Zhou, and J. Lu, "Dimension embeddings for monocular 3d object detection," in *CVPR*, 2022, pp. 1589–1598.
[56] J. Gu, B. Wu, L. Fan, J. Huang, S. Cao, Z. Xiang, and X.-S. Hua, "Homography loss for monocular 3d object detection," in *CVPR*, 2022, pp. 1080–1089.
[57] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012, pp. 3354–3361.
[58] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *CVPR*, 2020, pp. 2446–2454.
[59] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals for accurate object class detection," *NeurIPS*, vol. 28, 2015.
[60] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
[61] X. Liu, N. Xue, and T. Wu, "Learning auxiliary monocular contexts helps monocular 3d object detection," in *AAAI*, 2022, pp. 1810–1818.