

Weakly Supervised 3D Object Detection via Multi-Level Visual Guidance

Kuan-Chih Huang¹, Yi-Hsuan Tsai², and Ming-Hsuan Yang^{1,2}

¹University of California, Merced ²Google

Abstract. Weakly supervised 3D object detection aims to learn a 3D detector with lower annotation cost, e.g., 2D labels. Unlike prior work which still relies on few accurate 3D annotations, we propose a framework to study how to leverage constraints between 2D and 3D domains without requiring any 3D labels. Specifically, we employ visual data from three perspectives to establish connections between 2D and 3D domains. First, we design a feature-level constraint to align LiDAR and image features based on object-aware regions. Second, the output-level constraint is developed to enforce the overlap between 2D and projected 3D box estimations. Finally, the training-level constraint is utilized by producing accurate and consistent 3D pseudo-labels that align with the visual data. We conduct extensive experiments on the KITTI dataset to validate the effectiveness of the proposed three constraints. Without using any 3D labels, our method achieves favorable performance against state-of-the-art approaches and is competitive with the method that uses 500-frame 3D annotations. Code and models will be made publicly available.

Keywords: 3D Object Detection · Weakly-Supervised 3D Object Detection · Visual Guidance

1 Introduction

A key feature of autonomous systems is the ability to perceive and localize 3D objects accurately in the surrounding environments [4, 8–10, 12, 28], enabling the agent (e.g., vehicles) to make informed decisions and navigate safely. However, acquiring annotated 3D labels for training 3D object detection models poses challenges due to its cost and complexity, particularly when compared with the speed of labeling 2D boxes on visual data (e.g., 3–16 times slower, as shown in [33]). Hence, weakly supervised learning for 3D object detection has emerged as a practical approach to address the annotation bottleneck.

Recently, numerous approaches have been proposed to train 3D object detectors with reduced annotation requirements. FGR [33] proposes to generate 3D box candidates from the corresponding 2D boxes with frustum geometric relationships in a non-learning manner. WS3D [21, 22] annotates the center of objects in BEV as weak labels and uses a few 3D labels for model training with cylindrical constraints. Several approaches [16, 17] incorporate image and LiDAR information to learn the detectors jointly. Nevertheless, existing methods either

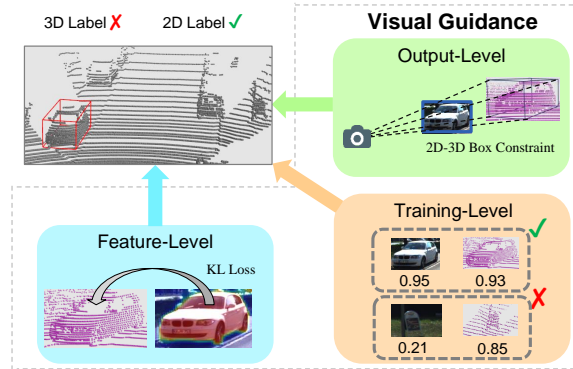


Fig. 1: Multi-level visual guidance for weakly-supervised 3D object detection. We propose a framework to learn a 3D object detector from weak labels, e.g., 2D bounding boxes on the image plane, using three different perspectives, including feature-, output-, and training-level constraints. Feature-level provides object-aware signals for point feature learning. Output-level incorporates 2D-3D box constraints to enforce the model to generate reasonable box prediction. Training-level guidance incorporates the confidence of 2D boxes into the pseudo-label technique to ensure the score consistency between 2D and 3D domains.

still require precise 3D box labels [16, 17, 21, 22], or they only achieve suboptimal performance [33] due to limited learning supervisions. Notably, none of these approaches explore the potential of incorporating various visual guidance across 2D and 3D during training.

In this work, we explore the integration of visual data into the training process of 3D object detectors, utilizing solely 2D annotations for weakly supervised 3D object detection, which is unique compared with the above-mentioned methods (see Table 1 for comparisons). As shown in Figure 1, we investigate the learning process with visual guidance from three perspectives: objectness learning from the feature level, response learning from the output level, and pseudo-label learning from the training level, with the following three observations.

Observation 1: Feature-level Guidance. For the well-calibrated image and LiDAR, the objectness predictions obtained from the image should align with the corresponding regions in the LiDAR data. For instance, when a point cloud is recognized as a foreground object by the 3D detector, its corresponding projected pixel on the image plane should align with a similar prediction made by the 2D detector, and vice versa. Hence, we utilize this idea on the feature level by mapping image features to point features and considering only the objectness part to enhance the feature learning process of 3D object detectors.

Observation 2: Output-level Guidance. We notice a substantial overlap between the 2D and projected 3D bounding boxes on the image plane. Based on this insight, we establish a distinct 2D-3D constraint to guide the supervision of 3D proposals. With this constraint, the model ensures that the estimated 3D box is accurately positioned within the frustum of the object’s image region to generate higher-quality proposals.

Category	Required 3D Label	Weak Label	Guided by Output	Guided by Feat.	Guided by Training
VS3D [26]	✗	Cls	✗	✗	✗
FGR [33]	✗	2D Box	✓	✗	✗
WS3D [22]	✓	BEV	✗	✗	✗
MTrans [16]	✓	2D Box	✓	✗	✗
VG-W3D (Ours)	✗	2D Box	✓	✓	✓

Table 1: Comparisons of VG-W3D and related work. Our approach is guided by multiple visual cues, including 2D box outputs, image features, and scores for training, without requiring 3D annotation.

Observation 3: Training-level Guidance. We find that using the initial 3D labels through a non-learning heuristic [33] can be noisy and partially missing objects due to the sparsity of point cloud data. Hence, it is crucial to refine these labels iteratively for higher accuracy. Another challenge is to reduce false positives of generated pseudo-labels since during the self-training process, the model may easily produce unexpected estimations with high confidence scores. Thus, we propose a solution by integrating prediction scores of 2D boxes from the visual domain into the pseudo-label technique to ensure score consistency for any object within both 2D and 3D domains.

Based on these observations, we propose a multi-level visual guidance approach for weakly supervised 3D object detection, named VG-W3D. We leverage visual cues to train a robust 3D object detector using only 2D annotations from three perspectives: feature, output, and training levels. Our method can further integrate with 2D annotations obtained from the off-the-shelf image object detectors trained from a different domain, making our method applicable to more scalable and cost-efficient 3D object detection.

Comprehensive experimental results validate the effectiveness of the proposed approaches. Specifically, when compared with methods in a similar annotation cost, our approach demonstrates a substantial improvement of at least 5.8% in AP_{3D} on the KITTI [6] dataset. Moreover, our method showcases comparable performance with state-of-the-art weakly supervised 3D detection methods that necessitate 500-frame 3D annotations.

2 Related Work

LiDAR-based 3D Object Detection. LiDAR-based 3D object detectors [7, 13, 28] have gained much attention recently. Depending on different point aggregation techniques, they can be roughly classified into voxel and point-based methods. Voxel-based methods [5, 20, 34, 36] voxelize the point cloud as voxel grid representations, followed by the 2D and 3D convolution operations for object detection. In [36], VoxelNet encodes the voxel features from the raw point cloud, followed by the dense region proposal network for 3D object detection. CIA-SSD [34] utilizes a lightweight network on the bird’s eye view (BEV) grid to learn spatial and semantic features and incorporates an IoU-aware confidence

refinement module for robust detection. Furthermore, Voxel-RCNN [20] applies voxel Region of Interest (RoI) pooling to extract voxel features within proposals for subsequent refinement, while VoxelNeXt [4] introduces a fully sparse architecture that directly predicts objects upon sparse voxel features instead of dense representations.

Numerous methods [11, 19, 28, 29] directly utilize the raw point clouds as inputs to extract point-level features. PointRCNN [28] proposes a two-stage framework that segments the foreground points, generates the object-wise proposal, and then refines the canonical coordinates. On the other hand, 3DSSD [19] introduces a fusion sampling technique using feature distance to ensure comprehensive information preservation. In [29], Point-GNN employs a graph neural network to create a more condensed representation of the point cloud data. Recently, Pointformer [24] uses a transformer module for local and global attention that operates directly on point clouds.

While these existing 3D object detectors have demonstrated desirable performance, they rely on 3D annotations for supervised learning, which is less efficient. In contrast, our approach emphasizes the development of a robust 3D detector without incurring labor-intensive costs.

3D Object Detection using Weak Labels. Annotating 3D bounding boxes on point clouds is time-consuming. As such, several methods focus on how to train a 3D detector with lower annotation costs to reduce the labor-intensive effort [16, 17, 21, 22, 26, 33]. VS3D [49] produces 3D proposals by analyzing point cloud density in an unsupervised manner. Then, points within boxes are projected onto the image plane, guided by a 2D model trained with class labels to determine whether the proposal contains an object. WS3D [21, 22] utilizes center clicks to annotate the coarse location of the objects in BEV for producing the initial cylindrical object proposals, followed by the refinement process at the second stage with few accurate 3D labels. In [33], FGR introduces a non-learning technique that utilizes 2D bounding boxes to identify frustum sub-point clouds and then calculates the most precise 3D bounding box based on the segmented point cloud using heuristic methods. MAP-Gen [17] and MTrans [16] utilize rich image data to address the sparsity challenge inherent in 3D point clouds. However, most methods mentioned above still require partial centroid or accurate 3D annotation to develop their approaches. In this work, we propose an approach that does not require any 3D annotations while still achieving competitive performance against existing weakly-supervised methods.

3 Proposed Approach

3.1 Framework Overview

Given a LiDAR point cloud \mathcal{P} and its corresponding camera image \mathcal{I} , our goal is to develop a 3D object detector without using any 3D object annotation. In this paper, we introduce VG-W3D, a multi-level visual guidance approach for weakly-supervised 3D object detection, aiming to learn precise 3D bounding boxes using dense visual signals and annotated 2D boxes from the image domain.

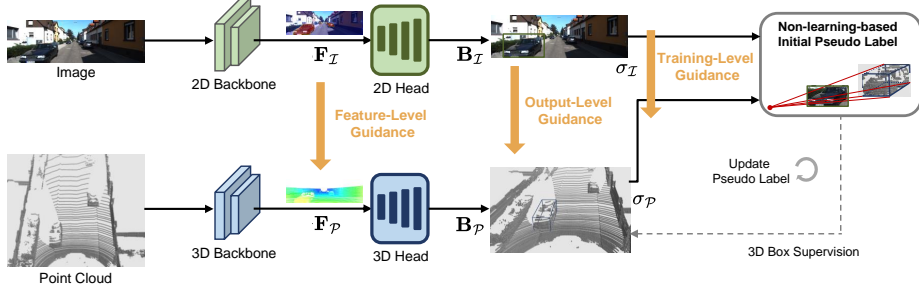


Fig. 2: Overall framework of the proposed VG-W3D. We utilize a non-learning method [33] to identify frustum point clouds of objects, followed by a heuristic algorithm to estimate the initial noisy bounding boxes (top right). In the image branch, we train an object detector based on 2D annotations to predict image features \mathbf{F}_I and 2D bounding boxes \mathbf{B}_I along with their confidence scores σ_I , which serve as visual guidance for training the 3D detector. Then, a PointNet-based 3D object detector is employed to extract point features \mathbf{F}_P and output 3D bounding boxes \mathbf{B}_P along with confidence scores σ_P . Our approach incorporates three levels of visual guidance for 3D training, namely feature-level (Section 3.2), output-level (Section 3.3), and training-level (Section 3.4). Note that the image branch is frozen during the training stage and is discarded during the inference stage.

As illustrated in Figure 2, VG-W3D utilizes feature-level, output-level, and training-level cues to guide 3D object detection. To obtain initial 3D boxes for training, we adopt a non-learning approach similar to FGR [33] to identify frustum point clouds of the objects, followed by a heuristic algorithm to estimate provisional 3D labels (see [33] for more details).

In the initial phase, we train a 2D detector [35] using the provided 2D box annotations \mathbf{B}_I , to extract visual features \mathbf{F}_I and predict 2D bounding boxes \mathbf{B}_I along with their corresponding confidence scores σ_I . Subsequently, we employ a 3D PointNet-based detector [28] to extract point cloud features \mathbf{F}_P and generate 3D bounding boxes \mathbf{B}_P with their detection scores σ_P .

In the following, we introduce three 2D visual cues to guide the 3D learning. First, the feature-level cues constrain the attention towards foreground objects in images and point clouds, ensuring consistent predictions between the two modalities (Section 3.2). Second, the output-level guidance ensures the logical positioning of predicted 3D proposals by enforcing the substantial overlaps between projected 3D boxes and 2D boxes on the image plane (Section 3.3). Finally, to produce high-quality pseudo-labels for retraining the 3D object detector, we incorporate image boxes and scores into the self-training technique, ensuring the improvement of pseudo-labels (Section 3.4).

3.2 Feature-Level Visual Guidance

For weakly-supervised 3D object detection, individual points usually lack clear guidance. Even when initial 3D labels are available [33], it is still challenging to

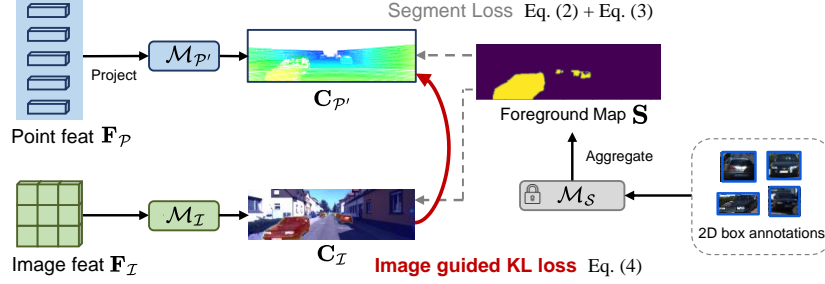


Fig. 3: Feature-level visual guidance. Once we acquire the projected point features $\mathbf{F}_{P'}$ and image feature \mathbf{F}_I , we utilize object foreground map \mathbf{S} to supervise the objections, in which the pretrained unsupervised instance segmentation module \mathcal{M}_S is applied to extract the object foreground map for each annotated 2D bounding box to generate \mathbf{S} . In addition, an image-guided KL divergence loss is applied to learn the distribution from the image features.

learn good 3D representations since these 3D labels are sometimes incomplete and noisy. Hence, we utilize dense visual hints to guide feature learning with known pixel-point mapping.

Feature Mapping. As shown in Figure 3, we consider image features $\mathbf{F}_I \in \mathbb{R}^{H \times W \times C}$ ($H \times W$ and C denote the feature size and the number of feature channels) and point cloud features $\mathbf{F}_P \in \mathbb{R}^{P \times C}$, with P being the number of points. Initially, we project the point cloud features onto the image plane using the camera calibration parameters, yielding the projected point features: $\mathbf{F}_{P'} = \text{Proj}(\mathbf{F}_P) \in \mathbb{R}^{H \times W \times C}$. Consequently, we obtain the projected point cloud features matching pixel points on the image plane for better feature learning.

Feature Guidance. One straightforward approach is enforcing point cloud features to mimic the image representations [14] that learn the features of image modality from the LiDAR modality with L2 loss:

$$\mathcal{L}_{feat} = \frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \|\mathbf{F}_I(i) - \mathbf{F}_{P'}(i)\|_2, \quad (1)$$

where \mathcal{A} is the valid pixel region on the image that matches with the projected points and $\|\cdot\|_2$ is the L_2 norm distance. However, this may harm the procedure of point cloud feature learning since the image features cannot provide more geometric information than point clouds. Instead, we propose to enforce the predicted objectness probability for both image and point cloud features, \mathbf{F}_I and $\mathbf{F}_{P'}$, to ensure that the 3D detector can recognize the foreground points with feature-level guidance.

We employ the segmentation map for objectness supervision to allow guidance only in object regions. Without incurring additional annotation costs, we utilize self-supervised segmentation methods [23, 32] to generate the foreground map without annotations. Specifically, within each ground truth 2D bounding box, objects are extracted along with their foreground maps for each object

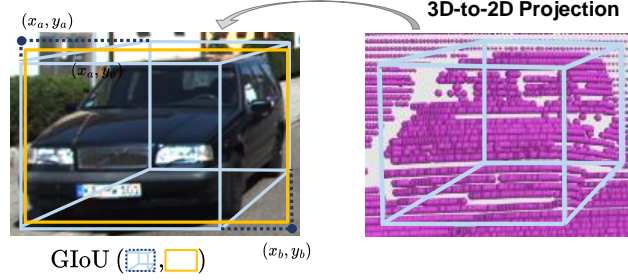


Fig. 4: Output-level visual guidance. The overlap of the projected 3D bounding box with the corresponding 2D bounding box signifies that 2D boxes can serve as supervision signals without 3D annotations. We use GIoU loss to constrain the projected box of the learned 3D box on the image plane with the ground truth 2D object box.

obtained by DINO [2]. These individual maps are then merged to form a segmentation ground truth map $\mathbf{S} \in \mathbb{R}^{H \times W}$. We then exploit a classifier $\mathcal{M}_{\mathcal{P}'}$ to map the point cloud features to the binary probabilities that predict the objectness of the projected point cloud: $\mathbf{C}_{\mathcal{P}'} = \mathcal{M}_{\mathcal{P}'}(\mathbf{F}_{\mathcal{P}'})$ via the focal loss FL:

$$\mathcal{L}_{seg}^{\mathcal{P}} = \frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \text{FL}(\mathbf{C}_{\mathcal{P}'}(i), \mathbf{S}(i)). \quad (2)$$

On the other hand, for the image domain, we add another branch on the 2D image detector with a linear classifier $\mathcal{M}_{\mathcal{I}}$ to learn the objectness for the pixels on the image: $\mathbf{C}_{\mathcal{I}} = \mathcal{M}_{\mathcal{I}}(\mathbf{F}_{\mathcal{I}})$, using a similar loss function to (2):

$$\mathcal{L}_{seg}^{\mathcal{I}} = \frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \text{FL}(\mathbf{C}_{\mathcal{I}}(i), \mathbf{S}(i)). \quad (3)$$

To this end, we can utilize KL divergence loss to enforce the objectness in point cloud modality and learn the similar distribution as image modality without losing the geometric information from point clouds:

$$\mathcal{L}_{kl} = \text{KL}(\mathbf{C}_{\mathcal{I}} \parallel \mathbf{C}_{\mathcal{P}'}). \quad (4)$$

3.3 Output-Level Visual Guidance

It is worth mentioning that any object detected with both 2D and 3D bounding boxes should exhibit a high overlap [25, 31], as illustrated in Figure 4. This implies that in weakly-supervised learning scenarios where the 3D bounding box is unknown, we can utilize the ground truth 2D box to supervise the 3D boxes predicted by the 3D detector.

First, given the predicted 3D bounding box $\mathbf{B}_{\mathcal{P}}$, we obtain its eight corners in 3D coordinates denoted as $\mathcal{C}_3(\mathbf{B}_{\mathcal{P}}) \in \mathbb{R}^{8 \times 3}$. Then, we obtain projected corners $\mathcal{C} \in \mathbb{R}^{8 \times 2}$ in 2D by using the known camera calibration parameters. Next, the bounding box of \mathcal{C} can be determined by:

$$(x_a, y_a) = \min(\mathcal{C}), (x_b, y_b) = \max(\mathcal{C}), \quad (5)$$

Algorithm 1: Training-level Guidance

Input : Point cloud \mathcal{P} , initial 3D box $\hat{\mathbf{B}}_0$, pre-trained image detector $\Theta_{\mathcal{I}}$ with 2D box annotations $\hat{\mathbf{B}}_{\mathcal{I}}$, and predicted confidence $\sigma_{\mathcal{I}}$;

Output: 3D bounding box predictions $\mathbf{B}_{\mathcal{P}}$

```

1 repeat
2   Train a 3D detector  $\Theta$  with  $\hat{\mathbf{B}}_t$ ,  $\hat{\mathbf{B}}_{\mathcal{I}}$  and  $\Theta_{\mathcal{I}}$ 
3   Obtain 3D pseudo-labels  $\mathbf{B}_{t+1}$  with confidence  $\sigma_{\mathcal{P}}$ 
4   Update pseudo-labels  $\mathbf{B}_{t+1}$  to  $\hat{\mathbf{B}}_{t+1}$  with:
      1)  $\mathbf{B}_{\text{overlap}} = \{\mathbf{B}_{t+1} | \text{IoU}(\text{Proj}(\mathbf{B}_{t+1}), \hat{\mathbf{B}}_{\mathcal{I}}) > \alpha_0, (\sigma_{\mathcal{P}} + \sigma_{\mathcal{I}})/2 > \alpha_1\}$ 
      2)  $\mathbf{B}_{\text{unmatch}} = \mathbf{B}_{t+1} \setminus \mathbf{B}_{\text{overlap}}; \mathbf{B}_{\text{score}} = \{\mathbf{B}_{\text{unmatch}} | \sigma_{\mathcal{P}} > \alpha_2\}$ 
      3)  $\hat{\mathbf{B}}_{t+1} = \mathbf{B}_{\text{overlap}} + \mathbf{B}_{\text{score}}$ 
5    $\mathbf{B}_{\mathcal{P}} = \hat{\mathbf{B}}_{t+1}$ 
6 until convergence;
```

where (x_a, y_a) and (x_b, y_b) are the top-left and bottom-right coordinates of the box \mathbf{B}_{proj} . As a result, we utilize the corresponding 2D box prediction $\mathbf{B}_{\mathcal{I}}$ to constraint the difference between these two boxes:

$$\mathcal{L}_{\text{box}} = \hat{\sigma}_{\mathcal{I}}(1 - \text{GIoU}(\mathbf{B}_{\mathcal{I}}, \mathbf{B}_{\text{proj}})), \quad (6)$$

where GIoU is the generalized intersection over union [27] to guide box learning. Compared with the normal IoU loss, GIoU loss can better alleviate the vanishing gradient issue of the non-overlapping case [27] between the projected 3D box and the ground truth 2D box.

In addition, $\hat{\sigma}_{\mathcal{I}} = \sigma_{\mathcal{I}} / \sum_i^N \sigma_{\mathcal{I}_i}$ is the normalized score for each predicted 2D box across all N objects in the same scene. We observe that objects with low confidence by the 2D detector indicate the uncertainty of the predicted box, which may not be a high-quality box. Therefore, it is not suitable to treat each box equally. As such, we introduce the prediction score $\hat{\sigma}_{\mathcal{I}}$ as the weight of the loss in (6) for each box. To this end, the proposed output-level guidance can ensure precise alignment between the projected bounding box and its 2D counterpart.

3.4 Training-Level Visual Guidance

One common approach to providing direct supervisory signals is to use pseudo-labels. We initially consider other approaches like [33] to produce 3D pseudo-labels without learning. However, they are prone to noise and may miss numerous objects. For instance, only about 2700 frames contain pseudo-labels among the training dataset (3712 frames) generated by [33]. Moreover, pseudo-labels may introduce additional false positives that may negatively affect self-training. To handle these issues, we introduce an image-guided approach to generate high-quality pseudo-labels, outlined in Algorithm 1. The method starts with initial 3D pseudo-labels $\hat{\mathbf{B}}_0$ from [33], 2D box annotation $\mathbf{B}_{\mathcal{I}}$, and a pre-trained 2D

object detector $\Theta_{\mathcal{I}}$. We utilize a 2D detector to predict the confidence score $\sigma_{\mathcal{I}}$ for each 2D annotation. In practice, we extract the confidence of the center index for each ground truth object from the heatmap predicted by the image detector.

Each iteration consists of three main steps. First, we train the 3D object detector based on pseudo-labels $\hat{\mathbf{B}}_t$ at round t and use the feature- and output-levels guidance as mentioned in Section 3.2 and 3.3, respectively. Second, we generate the initial pseudo-labels $\hat{\mathbf{B}}_{t+1}$ along with corresponding confidence $\sigma_{\mathcal{P}}$ for the next round. Finally, we filter pseudo-labels based on the following criteria to ensure the quality: 1) Utilizing the Hungarian algorithm, we match 2D and projected 3D bounding boxes, retaining those with an IoU score greater than the threshold α_0 . Moreover, the averaged confidence score from 2D and 3D boxes should be larger than α_1 . The generated pseudo-labels are denoted as $\mathbf{B}_{\text{overlap}}$. 2) For the remaining projected 3D boxes $\mathbf{B}_{\text{unmatch}} = \mathbf{B}_{t+1} \setminus \mathbf{B}_{\text{overlap}}$, we apply the non-maximum suppression (NMS) to eliminate redundant boxes, retaining only those with high confidence α_2 , termed as $\mathbf{B}_{\text{score}}$. The final selected 3D pseudo-labels are the two sets: $\hat{\mathbf{B}}_{t+1} = \mathbf{B}_{\text{overlap}} + \mathbf{B}_{\text{score}}$.

3.5 Training Objectives

We apply two loss functions to train our weakly-supervised 3D object detection framework. For scenes with the 3D pseudo-labels:

$$\mathcal{L}_{pl} = \mathcal{L}_{rpn} + \mathcal{L}_{rcnn} + \mathcal{L}_{seg}^{\mathcal{P}} + \mathcal{L}_{kl}, \quad (7)$$

where \mathcal{L}_{rpn} and \mathcal{L}_{rcnn} represent the loss functions of RPN and refinement network proposed in [28], respectively. $\mathcal{L}_{seg}^{\mathcal{P}}$ and \mathcal{L}_{kl} are the feature-level guidance as in (2) and (4). For scenes with only 2D labels, the weakly-supervised loss functions are applied:

$$\mathcal{L}_{weak} = \mathcal{L}_{seg}^{\mathcal{P}} + \mathcal{L}_{kl} + \mathcal{L}_{box}, \quad (8)$$

where \mathcal{L}_{box} is the loss of box-level guidance defined in (6). For the 2D object detector, we adopt the same loss function as CenterNet [35] with an additional loss $\mathcal{L}_{seg}^{\mathcal{I}}$ in (3).

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate the proposed method on the KITTI 3D object detection [6] dataset, which comprises 7481 images for training and 7518 images for testing. Following the protocol established in [3], we divide the training samples into a training set (3712) and a validation set (3769) for conducting experiments for the validation set and ablation studies.

Evaluation Metrics. We assess performance using the average precision (AP) metric for 3D object detection and bird’s eye view (BEV) detection tasks. To

Method	Weak Labels	Required 3D Annotations	AP _{3D} @IoU=0.7			AP _{BEV} @IoU=0.7		
			Easy	Mod.	Hard	Easy	Mod.	Hard
PointPillars [13]	-	✓	82.58	74.31	68.99	90.07	86.56	82.81
PointRCNN [28]	-	✓	86.96	75.64	70.70	92.13	87.39	82.72
WS3D [22]	BEV Centroid	534o	80.15	69.64	63.71	90.11	84.02	76.97
WS3D (2021) [21]	BEV Centroid	534o	80.99	70.59	64.23	90.96	84.93	77.96
MAP-Gen [17]	2D Box	500f	81.51	74.14	67.55	90.61	85.91	80.58
MTrans (PointPillars) [16]	2D Box	500f	77.65	67.48	62.38	-	-	-
MTrans (PointRCNN) [16]	2D Box	500f	83.42	75.07	68.26	91.42	85.96	78.82
FGR [33]	2D Box	✗	80.26	68.47	61.57	90.64	82.67	75.46
VG-W3D (Ours)	2D Box	✗	84.09	74.28	67.90	91.88	85.61	80.67

Table 2: Detection performance of the Car category on the KITTI test set. We compare our VG-W3D with fully-supervised baselines (first block), weakly-supervised methods that require accurate 3D annotations (second block), and weakly-supervised baselines without any 3D annotation (last block). 500f and 534o denote that the methods require annotating 500 frames and 534 objects (about 120 frames) for training, respectively.

mitigate bias [30], we adopt the 40 recall positions metric (AP₄₀) instead of the original 11 (AP₁₁). The benchmark categorizes detection difficulty into three levels ("Easy," "Mod.," "Hard"), considering size, occlusion, and truncation. All methods are ranked based on AP_{3D} score in the moderate setting (Mod.), aligning with the KITTI benchmark.

Implementation Details. We adopt CenterNet [35] as the 2D detector trained for 140 epochs with an extra head for foreground map segmentation in (3). On the other hand, we utilize PointRCNN [28] as the 3D detector for fair comparisons, trained for 30 epochs. For output-level guidance, we choose the overlapped bounding box with IoU larger than 0.5 for training. For training-level guidance, we set $\alpha_0 = 0.5$, $\alpha_1 = 0.5$ and $\alpha_2 = 0.95$. We will make the source codes and models available to the public.

4.2 Main Results

KITTI test set. Table 2 compares our VG-W3D with several state-of-the-art weakly-supervised 3D object detection methods and fully-supervised baselines on the KITTI test set. Compared with the fully-supervised baseline, PointRCNN, our VG-W3D achieves competitive performance without using any 3D annotations, demonstrating the effectiveness of our method. Furthermore, compared with our weakly-supervised baseline, FGR [33], our approach obtains improvements by **3.83/5.81/6.33** in AP_{3D} at IoU threshold 0.7 on three settings, which show the effectiveness of the proposed visual guidance approach. Furthermore, compared with methods (e.g., WS3D [21, 22]) that require 3D or BEV central annotations, our method performs favorably against these approaches.

KITTI val set. We also conduct experiments on the KITTI validation set in Table 3. Similarly, our approach achieves favorable performance against several

Method	Required 3D Annotations	AP _{3D} @IoU=0.7		
		Easy	Mod.	Hard
PointPillars [13]	✓	86.10	76.58	72.79
PointRCNN [28]	✓	88.99	78.71	78.21
WS3D [22]	534o	84.04	75.10	73.29
WS3D (2021) [21]	534o	85.04	75.94	74.38
MAP-Gen [17]	500f	87.87	77.98	76.18
MTrans [16]	125f	87.64	77.31	74.32
MTrans [16]	500f	88.72	78.84	77.43
VS3D [26]	✗	9.09	5.73	5.03
WS3DPR [18]	✗	60.01	44.48	36.93
FGR [33]	✗	86.68	73.55	67.91
FGR [33]*	✗	87.19	74.00	68.34
VG-W3D (Ours)	✗	91.32	78.89	74.70

Table 3: Detection performance of the Car category on the KITTI validation set. We compare our VG-W3D with fully-supervised and weakly-supervised methods. * means the results are reproduced by ourselves.

	Feature	Output	Training	AP _{3D} @IoU=0.7		
				Easy	Mod.	Hard
(a)	✗	✗	✗	87.19	74.00	68.34
(b)	✓	✗	✗	89.12	74.29	70.78
(c)	✗	✓	✗	88.95	76.42	71.58
(d)	✗	✗	✓	88.95	77.75	73.31
(e)	✓	✓	✗	89.43	76.71	72.06
(f)	✓	✓	✓	91.32	78.89	74.70

Table 4: Ablation study with different levels of visual guidance in our proposed framework on the KITTI validation set.

weakly-supervised methods. Specifically, compared to MTrans [16] that requires 500-frame 3D annotations for training, our VG-W3D with the same 3D detector (i.e., PointRCNN [28]) achieves better results on most metrics, which validates the effectiveness of our approach. Furthermore, compared with VS3D [26] and WS3DPR [18] that only utilize class-level labels for classifying the proposal, our approach can effectively guide the learning from multiple perspectives, resulting in significant performance improvement.

4.3 Ablation Study and Analysis

Importance of different levels of visual guidance. In Table 4, we show the effectiveness of different levels of visual guidance. Without any proposed visual guidance, the baseline (a) we reproduce via FGR [33] only achieves undesirable performance since the initial 3D annotations from the non-learning method are noisy. After adding the training-level constraint in row (b), the model is able to learn the objectness of the points. Utilizing the proposed output-level guidance in row (c) can further enforce the model to generate reasonable proposals to enhance detection ability. We also show the benefit of retraining the model with

Loss	Mask	AP _{3D} @IoU=0.7		
		Easy	Mod.	Hard
X	Segment	88.60	74.61	71.10
L2	Segment	88.87	75.69	71.34
KL	2D Box	89.13	74.65	69.88
KL	Segment	89.43	76.71	72.06

Table 5: Ablation study with different designs of feature-level visual guidance on the KITTI validation set.

2D-3D loss	AP _{3D} @IoU=0.7			AP _{BEV} @IoU=0.7		
	Easy	Mod.	Hard	Easy	Mod.	Hard
L1	88.81	74.70	71.36	94.56	85.05	80.84
IoU	88.51	74.50	71.40	95.16	85.71	82.15
GIoU	89.43	76.71	72.06	95.51	86.31	81.84

Table 6: Ablation study with different 2D-3D constraint loss of our output-level visual guidance on the KITTI validation set.

our training-level guidance in row (d). Finally, the results in rows (e) and (f) demonstrate the efficacy of combining visual guidance from feature-, output-, and training-levels.

Designs in feature-level guidance. We investigate the effectiveness of the proposed feature-level guidance in Table 5¹. In the first row, a performance drop is observed when the objectness probability learned from the image domain is omitted. In addition, substituting the KL divergence loss with L2 loss and attempting to mimic features instead of objectness prediction yields unsatisfactory results, aligning with our expectations outlined in Section 3.2. Finally, using the 2D bounding box as a mask for foreground supervision is not effective, as it fails to provide valuable signals for point clouds, e.g., certain regions inside the box might belong to the background.

Designs in output-level guidance. In Table 6, various options for output-level guidance are compared. Instead of regressing the IoU between 2D and 3D bounding boxes, we can utilize the L1 loss to learn the corners of the 3D box predicted from the pre-trained image detector. However, this approach is less effective due to the noisy nature of initial 3D box annotations, impacting the ability of the image detector to learn the corners as guidance. Furthermore, our results indicate that the Generalized Intersection over Union (GIoU) loss yields better performance than the Intersection over Union (IoU) loss, since the projected 3D box often does not perfectly overlap with the 2D box.

Designs in training-level guidance. We explore the effectiveness of various training-level guidance designs in Table 7. Initially, we employ the pseudo-labeling (PL) technique to re-generate labels using the trained model. However,

¹ To better understand the effect of feature- and output-level guidance, we do not include the training-level guidance that involves multiple retraining iterations in the ablation studies.

Method	AP _{3D} @IoU=0.7			AP _{BEV} @IoU=0.7		
	Easy	Mod.	Hard	Easy	Mod.	Hard
PL	91.06	76.64	73.79	95.24	86.06	84.32
Overlap	91.07	77.55	73.74	95.21	86.01	84.28
Overlap + Score	91.31	78.89	74.70	95.50	88.16	85.99

Table 7: Ablation study with different designs of training-level guidance on the KITTI validation set. “PL” denotes the pseudo-labeling technique, while “Overlap” and “Score” are $\mathbf{B}_{\text{overlap}}$ and $\mathbf{B}_{\text{score}}$ in Algorithm 1.

PL	Recall (IoU=0.5)	Recall (IoU=0.7)	Method	2D Anno.	AP _{3D} @IoU=0.7		
					Easy	Mod.	Hard
Initial	0.5422	0.4671	FGR [33]	KITTI	86.68	73.55	67.91
Round 1	0.8492	0.7192	VG-W3D (Ours)	KITTI	91.32	78.89	74.70
Round 2	0.8789	0.7422	FGR [33]	COCO	88.08	73.87	68.69
Round 3	0.8795	0.7421	VG-W3D (Ours)	COCO	89.12	78.21	74.21

Table 8: Quality of the pseudo-label on the KITTI validation set. PL notes the Pseudo Label.

Table 9: Integration with the off-the-shelf 2D object detector, which is pre-trained on a different dataset.

this approach yields less favorable results as the model generates numerous false positives, causing some negative effects in the training process. Adding the constraints of overlapping the projected 3D box and ground truth 2D box ($\mathbf{B}_{\text{overlap}}$ in Algorithm 1) slightly helps self-training. Finally, integrating two of our training-level guidance (i.e., $\mathbf{B}_{\text{overlap}} + \mathbf{B}_{\text{score}}$ in Algorithm 1) preserves predictions with high confidence scores, enables us to identify more objects lacking annotations in the image domain. This significantly improves the pseudo-labeling process.

Quality of pseudo-labels. In Table 8, we investigate the quality of the initial pseudo-label [33] and our proposed pseudo-label approach. The 3D labels generated through the non-learning approach often contain noise and missing information, particularly when the frustum point cloud lacks clear separation to locate the bounding box accurately. Therefore, the recall only achieves 46.71% at IoU=0.7. After the first round of training with our approach, the quality of the pseudo-label is improved by approximately 25% recall, showcasing the effectiveness of our approach. Finally, the recall improvement of the pseudo label saturates after the second-round training.

Cooperate with pre-trained 2D object detector. Our approach is able to cooperate with existing pre-trained off-the-shelf 2D object detectors, as shown in Table 9. With the DETR [1] object detector trained on the COCO dataset [15], we can directly use it to detect the objects on the KITTI dataset and generate 2D boxes. Initially, we generate 3D bounding boxes using [33] and train our model, establishing baseline results. Subsequently, we apply our proposed visual guidance approach. It is worth noting that our method achieves competitive performance compared to using 2D annotations. This demonstrates the potential scalability and effectiveness of the proposed framework.

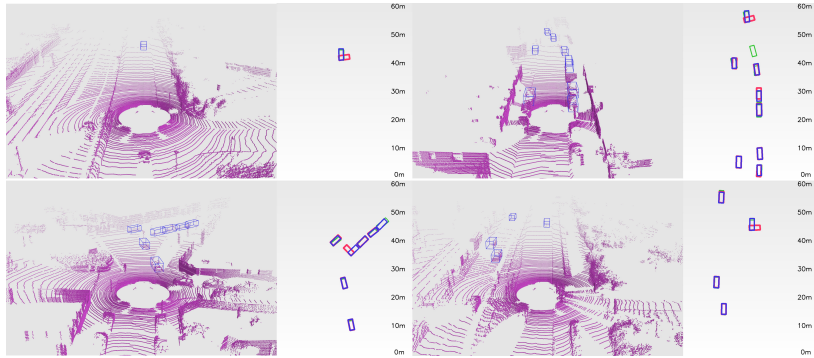


Fig. 5: Qualitative visualizations on the KITTI validation set. We provide the predictions on the LiDAR view (left) and bird’s eye view (right) in each result. The purple boxes in the LiDAR data and BEV plane indicate the predictions from our VG-W3D. The green and pink boxes on BEV are the ground truth and predictions from the FGR [33] baseline (without the proposed visual guidance), respectively. We show that our predictions have a closer alignment with the ground truth, while FGR often fails to enclose the point cloud tightly.

4.4 Qualitative Results

We show qualitative examples of the KITTI validation set in Figure 5. In the results, we show predictions of our method both on the LiDAR and BEV (denoted as purple boxes), while results of FGR [33] and ground truths on BEV are denoted as pink and green boxes, respectively.

Compared to FGR [33] which is only trained with initial 3D labels, the predictions from our VG-W3D show a closer alignment with the ground truth. This validates the efficacy of the three proposed visual guidances in enhancing object detection performance. In addition, we observe that the non-learning approach typically fails to enclose the point cloud tightly, resulting in labels with inaccurate heading and consequently producing incorrect guidance.

5 Conclusions

In this paper, we introduce a multi-level visual guidance approach for weakly-supervised 3D object detection by only using 2D annotations. Leveraging rich information from the image domain, our method employs visual data to guide the 3D learning process from feature-, output-, and training-level perspectives. At the feature level, our guidance ensures that point cloud features predict objectness distributions similar to those of the image detector. Output-level guidance enforces the LiDAR model to generate proposals located at reasonable positions. Moreover, training-level guidance generates high-quality 3D labels consistent with 2D bounding boxes, facilitating the retraining process. Extensive experiments conducted on the KITTI dataset demonstrate that our method outperforms state-of-the-art approaches, achieving competitive performance without needing any 3D annotations.

Acknowledgements. This work was supported in part by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/ Interior Business Center (DOI/IBC) contract number 140D0423C0074. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or the U.S. Government.

References

1. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European Conference on Computer Vision (ECCV) (2020) [13](#)
2. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: IEEE International Conference on Computer Vision (ICCV) (2021) [7](#)
3. Chen, X., Kundu, K., Zhu, Y., Berneshawi, A.G., Ma, H., Fidler, S., Urtasun, R.: 3d object proposals for accurate object class detection. In: Advances in Neural Information Processing Systems (NeurIPS) (2015) [9](#)
4. Chen, Y., Liu, J., Zhang, X., Qi, X., Jia, J.: Voxelnex: Fully sparse voxelnet for 3d object detection and tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023) [1](#), [4](#)
5. Deng, J., Shi, S., Li, P., Zhou, W., Zhang, Y., Li, H.: Voxel r-cnn: Towards high performance voxel-based 3d object detection. AAAI Conference on Artificial Intelligence (AAAI) (2021) [3](#)
6. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2012) [3](#), [9](#)
7. He, C., Zeng, H., Huang, J., Hua, X.S., Zhang, L.: Structure aware single-stage 3d object detection from point cloud. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020) [3](#)
8. Huang, K.C., Huang, Y.K., Hsu, W.H.: Multi-stream attention learning for monocular vehicle velocity and inter-vehicle distance estimation. In: British Machine Vision Conference (BMVC) (2021) [1](#)
9. Huang, K.C., Lyu, W., Yang, M.H., Tsai, Y.H.: Ptt: Point-trajectory transformer for efficient temporal 3d object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024) [1](#)
10. Huang, K.C., Wu, T.H., Su, H.T., Hsu, W.H.: Monodtr: Monocular 3d object detection with depth-aware transformer. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022) [1](#)
11. Koo, I., Lee, I., Kim, S.H., Kim, H.S., Jin Jeon, W., Kim, C.: Pg-rcnn: Semantic surface point generation for 3d object detection. In: IEEE International Conference on Computer Vision (ICCV) (2023) [4](#)
12. Kuan-Chih Huang, M.H.Y., Tsai, Y.H.: Delving into motion-aware matching for monocular 3d object tracking. In: IEEE International Conference on Computer Vision (ICCV) (2023) [1](#)

13. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: Pointpillars: Fast encoders for object detection from point clouds. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019) [3](#), [10](#), [11](#)
14. Li, Y., Chen, Y., Qi, X., Li, Z., Sun, J., Jia, J.: Unifying voxel-based representation with transformer for 3d object detection. In: Advances in Neural Information Processing Systems (NeurIPS) (2022) [6](#)
15. Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P.: Microsoft coco: Common objects in context. arXiv preprint arXiv:1405.0312 (2014) [13](#)
16. Liu, C., Qian, X., Huang, B., Qi, X., Lam, E., Tan, S.C., Wong, N.: Multimodal transformer for automatic 3d annotation and object detection. In: European Conference on Computer Vision (ECCV) (2022) [1](#), [2](#), [3](#), [4](#), [10](#), [11](#)
17. Liu, C., Qian, X., Qi, X., Lam, E.Y., Tan, S.C., Wong, N.: Map-gen: An automated 3d-box annotation flow with multimodal attention point generator. In: ICPR (2022) [1](#), [2](#), [4](#), [10](#), [11](#)
18. Liu, H., Wang, H.M.Y., Zou, B., Hu, T., Wang, R., Chen, J.: Eliminating spatial ambiguity for weakly supervised 3d object detection without spatial labels. ACM Multimedia (2022) [11](#)
19. Luo, S., Dai, H., Shao, L., Ding, Y.: M3dssd: Monocular 3d single stage object detector. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021) [4](#)
20. Mao, J., Xue, Y., Niu, M., et al.: Voxel transformer for 3d object detection. IEEE International Conference on Computer Vision (ICCV) (2021) [3](#), [4](#)
21. Meng, Q., Wang, W., Zhou, T., Shen, J., Jia, Y., Van Gool, L.: Towards a weakly supervised framework for 3d point cloud object detection and annotation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2020) [1](#), [2](#), [4](#), [10](#), [11](#)
22. Meng, Q., Wang, W., Zhou, T., Shen, J., Van Gool, L., Dai, D.: Weakly supervised 3d object detection from lidar point cloud. In: European Conference on Computer Vision (ECCV) (2020) [1](#), [2](#), [3](#), [4](#), [10](#), [11](#)
23. Niu, D., Wang, X., Han, X., Lian, L., Herzig, R., Darrell, T.: Unsupervised universal image segmentation (2023) [6](#)
24. Pan, X., Xia, Z., Song, S., Li, L.E., Huang, G.: 3d object detection with point-former. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021) [4](#)
25. Park, J.D., Xu, C., Zhou, Y., Tomizuka, M., Zhan, W.: Detmatch: Two teachers are better than one for joint 2d and 3d semi-supervised object detection. In: European Conference on Computer Vision (ECCV) (2022) [7](#)
26. Qin, Z., Wang, J., Lu, Y.: Weakly supervised 3d object detection from point clouds. ACM Multimedia (2020) [3](#), [4](#), [11](#)
27. Rezatofghi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: A metric and a loss for bounding box regression. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019) [8](#)
28. Shi, S., Wang, X., Li, H.: Pointcnn: 3d object proposal generation and detection from point cloud. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019) [1](#), [3](#), [4](#), [5](#), [9](#), [10](#), [11](#)
29. Shi, W., Rajkumar, R.R.: Point-gnn: Graph neural network for 3d object detection in a point cloud. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020) [4](#)

30. Simonelli, A., Bulò, S.R.R., Porzi, L., López-Antequera, M., Kotschieder, P.: Disentangling monocular 3d object detection. In: IEEE International Conference on Computer Vision (ICCV) (2019) [10](#)
31. Tang, Y.S., Lee, G.H.: Transferable semi-supervised 3d object detection from rgb-d data. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019) [7](#)
32. Wang, X., Girdhar, R., Yu, S.X., Misra, I.: Cut and learn for unsupervised object detection and instance segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023) [6](#)
33. Wei, Y., Su, S., Lu, J., Zhou, J.: FGR: Frustum-Aware Geometric Reasoning for Weakly Supervised 3D Vehicle Detection. In: IEEE International Conference on Robotics and Automation (ICRA) (2021) [1](#), [2](#), [3](#), [4](#), [5](#), [8](#), [10](#), [11](#), [13](#), [14](#)
34. Zheng, W., Tang, W., Chen, S., Jiang, L., Fu, C.W.: Cia-ssd: Confident iou-aware single-stage object detector from point cloud. In: AAAI Conference on Artificial Intelligence (AAAI) (2021) [3](#)
35. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. In: arXiv preprint arXiv:1904.07850 (2019) [5](#), [9](#), [10](#)
36. Zhou, Y., Tuzel, O.: Voxelnet: End-to-end learning for point cloud based 3d object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018) [3](#)