Semi-Supervised Monocular 3D Object Detection by Multi-View Consistency

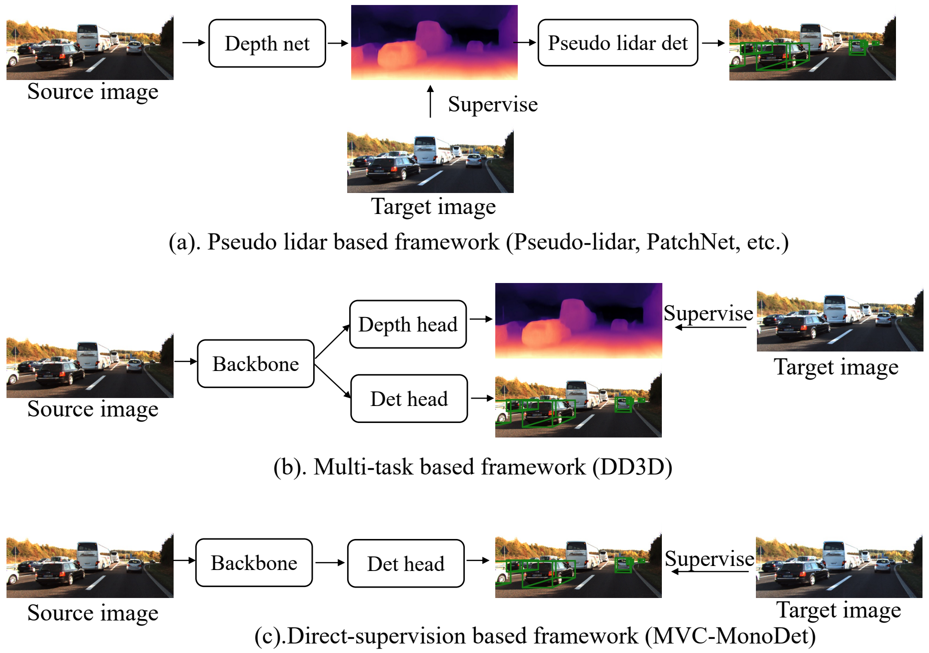
Qing Lian[1], Yanbo Xu[1], Weilong Yao[3], Yingcong-Chen[2,1], and Tong Zhang[1,4]
[1]The Hong Kong University of Science and Technology
[2]The Hong Kong University of Science and Technology (Guangzhou)
[3]Autowise.AI [4] Google Research
{qlianab, yxubu}@connect.ust.hk yaoweilong@autowise.ai
{yingcongchen, tongzhang}@ust.hk

Abstract. The success of monocular 3D object detection highly relies on considerable labeled data, which is costly to obtain. To alleviate the annotation effort, we propose MVC-MonoDet, the first semi-supervised training framework that improves **Mono**cular 3D object **det**ection by enforcing **m**ulti-**v**iew **c**onsistency. In particular, a box-level regularization and an object-level regularization are designed to enforce the consistency of 3D bounding box predictions of the detection model across unlabeled multi-view data (stereo or video). The box-level regularizer requires the model to consistently estimate 3D boxes in different views so that the model can learn cross-view invariant features for 3D detection. The object-level regularizer employs an object-wise photometric consistency loss that mitigates 3D box estimation error through structure-from-motion (SFM). A key innovation in our approach to effectively utilize these consistency losses from multi-view data is a novel relative depth module that replaces the standard depth module in vanilla SFM. This technique allows the depth estimation to be coupled with the estimated 3D bounding boxes, so that the derivative of consistency regularization can be used to directly optimize the estimated 3D bounding boxes using unlabeled data. We show that the proposed semi-supervised learning techniques effectively improve the performance of 3D detection on the KITTI and nuScenes datasets. We also demonstrate that the framework is flexible and can be adapted to both stereo and video data.

Keywords: Monocular 3D Object Detection, Semi-supervised Training, Structure From Motion

1 Introduction

Localizing objects in 3D space is an essential task in autonomous driving, which enables systems to perceive and understand surrounding environments. Motivated by the cheap and easy-to-deploy properties, academia and industry have been made a great effort to tackle monocular-based 3D object detection. Recently, deep learning based approaches [6, 54, 2, 20, 50, 29, 5] have achieved great success, leading to sophisticated deep neural networks as the main solution. The



(a). Pseudo lidar based framework (Pseudo-lidar, PatchNet, etc.)

(b). Multi-task based framework (DD3D)

(c).Direct-supervision based framework (MVC-MonoDet)

Fig. 1. Visualization of 3 frameworks in utilizing multi-view data to improve monocular 3D detection. (a) The pseudo-lidar based framework [44, 27] can use multi-view images to improve depth estimation model, leading to better image to lidar data conversion. (b) The multi-task framework (*e.g.,* DD3D [30]) builds a shared backbone for 3D detection and depth estimation. The multi-view data can be leveraged to train a stronger backbone by depth estimation. (c) Our MVC-MonoDet provides *direct* supervision signals for the detection model and no latent depth estimation module is required.

training of such neural networks often requires a large amount of high-quality labeled data. However, labeling 3D annotation is very tedious and expensive, as even humans can not directly annotate the ground-truth from a single image perfectly [13, 4].

Typically, semi-supervised learning is a promising direction to relieve the annotation burden. Existing approaches [49, 23, 39, 51, 41] have primarily focused on 2D tasks. However, recent work [29] identifies that the 3D detection performance is dominated by accurately regressing the 3D attributes (3D location, dimension and orientation). To improve the performance of the 3D attributes regression, we consider utilizing unlabeled multi-view data (stereo or video) to provide external 3D supervision. Meanwhile, the unlabeled multi-view data in autonomous driving scenarios is abundant and cheap to collect.

Existing monocular 3D detectors can utilize the multi-view data from two perspectives: data conversion in pseudo-lidar [44, 48] and shared representation in multi-task [30] frameworks. As visualized in Fig 1, these two frameworks require intermediate pixel-level depth representation to bridge the 3d detection with multi-view data. However, the objectives of these two tasks are different, where depth estimation focuses on background and object surface, but 3D detection only considers the object center. This difference may cause the supervision bias to the background regions and ignore the object-level 3D attributes. Furthermore, it is also demonstrated [11, 22] that neural networks learn different visual cues for these two tasks.

To better utilize the multi-view data, we design two kinds of multi-view consistency regularization that provide *direct* supervision signals on the foreground objects. (1) From the box space, we enforce the model to estimate consistent 3D bounding boxes in different views. This regularizes the model to learn robust features for different view angles and positions, leading to better generalization on the unlabeled and unseen data.

(2) From the object space, we design an object-wise photometric consistency module that utilizes structure-from-motion (SFM) to directly optimize 3D box. The vanilla SFM learner [52, 14] is tailored for depth estimation that leverages the photometric error between the source and projected views to represent and mitigate depth error. However, the standard depth estimation module in vanilla SFM is not coupled with 3D bounding boxes, and the corresponding SFM module can not be directly used to optimize 3D box positions. Inspired by Stereo R-CNN [19], we design a relative depth module that couples pixel-level depth with 3D boxes, so that the cross-view photometric consistency can be used to directly optimize the detection error. Based on this technique, we can directly mitigate the bounding boxes error by regularizing the cross-view photometric consistency.

We validate the effectiveness of our MVC-MonoDet on two standard 3D detection benchmarks: KITTI [13] and nuScenes [4] datasets. On the KITTI dataset, we show that the proposed approach can leverage stereo or video data to improve the state-of-the-art fully-supervised approaches with 22% and 11%. On the nuScenes dataset, we witness a relative 18% and 5% improvement with 10% and 100% of labeled data.

Our main contributions are as follows:

- We provide the first multi-view semi-supervised training framework for monocular 3D object detection. The framework leverages abundant and cheap unlabeled multi-view data to alleviate 3D annotation burden.
- Based on the multi-view framework, a box-level and an object-level consistency regularization are proposed to improve the 3D detector, and a relative depth module is proposed to allow effective coupling of 3D box error with the consistency losses.
- Experimental results on the KITTI and nuScenes datasets demonstrate the effectiveness of our semi-supervised learning framework with different types of multi-view data.

2 Related work

We briefly review the recent work on monocular 3D object detection, semi-supervised object detection and self-supervised learning with multi-view data.

2.1 Monocular 3D object detection

Traditional monocular 3D object detection methods [7, 54, 6, 2, 35, 38] recover the 3D bounding boxes by using the shape priors, semantic information, ground plane assumption, *etc*. To alleviate the challenging depth recovery, later work [38, 25, 8] pays more attention to the design of training pipelines [29, 25] and loss function [38]. Except for directly adopting neural networks to estimate depth, several studies [20, 21, 24, 5] propose to reason depth by solving the geometric constraints between 2D and 3D coordinates.

In addition, some approaches adopt pixel-level depth estimation models to assist the 3D detection. Pseudo-lidar based approaches [44, 28, 27] project the image and depth data into pseudo point cloud and adopt point cloud detectors [17, 34, 33] to localize objects. Except for projecting the input modality, D4LCN and it's follow-up [12, 42] leverage depth map to build dynamic convolution or graph propagation modules for better extracting 3D features in 2D space. DD3D [30] proposes a multi-task framework that leverages depth estimation to pre-train a strong feature representation for 3D object detection. By connecting depth estimation with 3D object detection, the large-scale unlabeled multi-view images can be utilized to improve the performance of 3D detection.

2.2 Semi-supervised object detection

Due to the heavy annotation burden in object detection, great efforts have been made to leverage unlabeled or weakly annotated data to improve performance. Inspired by the success of confidence regularization in semi-supervised classification, one line of approaches [15, 40, 31] focus on designing consistency regularization methods with different kinds of image perturbation. Another line of approaches [49, 23, 39, 51, 41] leverage neural networks to annotate pseudo labels for self-training. Despite the fast development in semi-supervised object detection, there is only one semi-supervised approach [21] for monocular 3D object detection. Li et al. [21] propose a consistency regularization method on a keypoint-constraint based approach [20], where the consistency regularization is employed on the intermediate keypoint detection task. However, due to the intermediate regularization, KM3D [21] only can be adopted to the keypoint-based approaches, while they are less effective compared to other end-to-end detectors [25, 50, 32]. By contrast, our work provides supervision signals on the final output, which is flexible and can be applied to arbitrary monocular 3D object detectors.

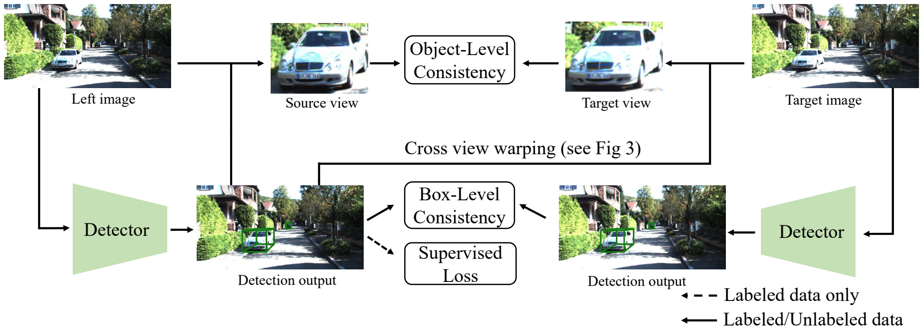


Fig. 2. Visualization of our multi-view semi-supervised training pipeline for monocular 3D object detection.

2.3 Self-supervised learning with multi-view data

It is a popular topic that trains a model to recover 3D information (*e.g.,* depth, ego-motion, flow, *etc.*) by unlabeled multi-view data. One group of methods take the multi-view consistency by structure from motion (SFM) to train neural networks for 3D reconstruction. Specifically, the supervision signal is obtained by pursuing photometric consistency between the origin frame and the reconstructed nearby frame, where the reconstructed nearby frame is warped by using the estimated depth and camera intrinsic [47, 52, 14, 1]. Traditional work [47] first takes the calibrated stereo camera to achieve the unsupervised depth training. Except for stereo data, video data is another cheap and easy-to-collect alternative for providing multi-view observations. However, the video data is unstructured, requiring further to estimate the ego and object poses. Zhou et al. [52] first design a unified framework that jointly trains a depth estimation and a camera motion model by minimizing the photometric error between the source and projected target frames. To address the scale inconsistent problem, Bian et al. [1] propose a geometric consistency loss to regularize the inconsistency prediction between adjacent views. Later studies further estimate object masks [14] or predict the object motion [18] to handle the occluded regions or dynamic objects.

However, little attention was paid to leveraging the multi-view information for monocular 3D object detection. One potential reason is that the cross-view warping in SFM requires the depth for each pixel surface, however, 3D detection only estimates the depth of object center. To mitigate this mismatch, we propose a relative depth module that recovers the per-pixel depth by object shape and estimated bounding boxes.

3 Background

Given an input image, the objectives of monocular 3D object detection are to recognize the interested objects and localize the corresponding 3D boxes. In our semi-supervised learning setting, we have a labeled split $\{I_s^i, I_t^i, T_{s\to t}^i, y^i\}_{i=1}^{N_l}$ with N_l labeled samples and unlabeled split $\{I_s^i, I_t^i, T_{s\to t}^i\}_{i=1}^{N_u}$ with N_u unlabeled samples, where I_s and I_t denote the multi-view image, and $T_{s\to t} \in R^{4\times4}$ denotes the ego-pose matrix for cross-view projection. In this paper, we use $u \in R^{1\times2}$ and $p \in R^{1\times3}$ to denote a point in 2D and 3D coordinates, respectively. $K \in R^{3\times3}$ denotes the camera intrinsic, and $I(p, K)$ represents the corresponding pixel indexed by point p. The label y comprises a set of 3D bounding boxes, which are represented by the eight corner points in the box: $b \in \mathcal{R}^{8\times3}$. In autonomous driving, the 3D bounding box can be further decomposed to object 3D location, dimension, and yaw angle.

The multi-view images can come from a stereo camera or a monocular camera with different time stamps (video). Our baseline model is the modified version of the one-stage detector CenterNet [54, 53] and adds several parallel heads for estimating the 3D attributes.

4 Approach

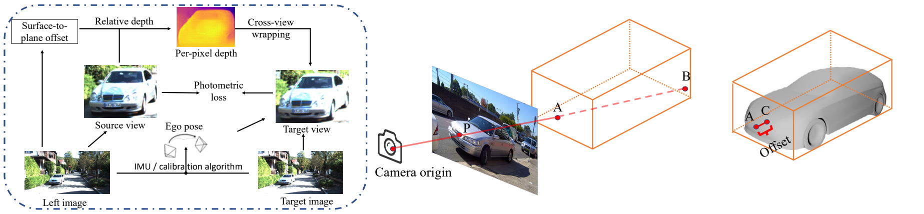
With unlabeled multi-view data, traditional approaches [52, 14] leverage multi-view consistency to train a per-pixel depth estimation network. However, as aforementioned, the supervision through the intermediate depth representation is not specialized for 3D detection, which may lead to sub-optimal utilization. Our framework provides two direct consistency regularization terms tailored for monocular 3D object detection. Figure 2 describes the overview of our multi-view semi-supervised training framework. Specifically, we introduce a box-level and an object-level consistency regularization techniques to a monocular 3D object detection model. From the box-level one, we regularize the model to estimate consistent 3D box attributes for the images taken from different views. This regularizes the model to be robust to variant view angles and positions. From the object-level one, we design an object-level photometric consistency loss to identify and mitigate bounding boxes error. It is worthy to note that during inference, only a single image is required to predict 3D bounding boxes.

4.1 Box-level Consistency

Given input images from different views, Box-Level Consistency (BLC) regularization enforces the estimated 3D boxes to be consistent in a rectified coordinate. This means that after converting the estimated 3D box to the target image, its attributes should match with the box estimated in the target image. Given the cross-view ego pose $T_{s\to t}$, the consistency loss is represented as:

$$\mathcal{L}_{output} = \frac{1}{N_b}\sum_{i=1}^{N_b} \|[\hat{b}_s^i, 1]T_{s\to t}^T - \hat{b}_t^i\|, \tag{1}$$

where $\hat{b}_s$ and $\hat{b}_t$ denote the boxes estimated in the source and target images, and N_b denotes the number of selected candidate boxes. In the video data, we further model the object motion [9] in the source to target conversion process.



(a). Pipeline of object-level photometric consistency module (b). Visualization of querying the relative depth (c). Visualization of the surface-to-plane offset

Fig. 3. (a). Visualization of the pipeline in computing the object-level photometric loss. The ego pose comes from pre-calibration (stereo) or the external hardware device and calibration algorithm (video). (b). A ray emitted from the camera origin to the pixel P in the image. It intersects with bounding box planes at points A and B. Point B is occluded by A (c). Points A and C lie on the bounding box planes and object surface, respectively.

In practice, one image may contain multiple objects. Therefore, matching the 3D boxes across images is necessary. In this paper, we propose a simple yet effective solution to achieve this. Popular 3D detection methods like MonoDLE [29] and CenterNet [54] produce 2D boxes in parallel with a 3D ones. The estimated 2D boxes are more accurate than the 3D boxes even with limited training data (See Appendix for illustration). Hence, we utilize the pixels in the region spanned by the 2D boxes to match 3D boxes. For each source box b_s^i, we calculate the SSIM [45] similarity scores with all the boxes in the target image and select the box that achieves the minimum SSIM scores as the paired target box b_t^i. To filter the background region, we filter out the bounding boxes that the estimated class confidence is smaller than 0.5.

4.2 Object-level Consistency

Note the box-level consistency regularization does not directly mitigate the prediction error but improves the performance through enhancing the model robustness. Furthermore, it can only provide sparse supervision. In this section, we propose an Object-Level Consistency (OLC) regularization to further leverage multi-view information for dense supervision.

The proposed OLC regularizes the photometric consistency between two views within bounding boxes. Specifically, OLC first utilizes SFM to reconstruct the source view of the objects by projecting the target views. Then we utilize the per-pixel photometric consistency to identify the bounding box prediction error. Note that the estimated bounding boxes are coupled with the depth used in the cross-view projection so that the consistency loss can reflect the localization error. Through enhancing the per-pixel photometric consistency, OLC provides much denser supervision than BLC regularization. The training procedure of one iteration is summarized as follows.

Step 1. We generate the source view by 3D boxes. On the labeled image, we directly take the ground truth. On the unlabeled images, we leverage the detector to predict the 3D boxes. We refer the source image area spanned by the 3D box as the source view. See the image patch of the source view as an example in Figure 3.

Step 2. We project the source view to the target view by the 3D boxes. Specifically, for each point in the source view, we calculate its projected location in the target view. The target image area spanned by the set of projected points is referred as the target view. Note that a source-to-target projection can not be done without knowing each pixel's depth in the source view. To this end, we design a *relative depth* with a *surface-to-cube offset head* to infer each pixel's depth through the 3D box. See details in Section 4.2.1.

Step 3. An object-level photometric loss is computed to measure the misalignment between the source and target views. To filter the noise in the view projection process, we also model the shape uncertainty to get an accurate photometric loss. See details in Section 4.2.2.

4.2.1 Target view projection by relative depth

This section presents the relative depth module used in Step 2, which infers each pixel's depth of the object's surface through a 3D box. To achieve this, we first start from a cube-shaped assumption that models all the objects as cube-shaped [19], and progressively learn the shape during training.

With a cube-shaped assumption, we can infer the depth of each pixel by a ray forwarding process [19] in a pinhole camera. Specifically, we first emit a ray from the camera origin o to the pixel in the source view p with vector $\vec{op}$ (see the solid red line in Figure 3.b for an example). Under the cube-shaped assumption, the pixel would be the perspective projection of a certain 3D point on the 3D box plane. In other words, the 3D point is the intersection between the ray and the 3D box planes. The intersections can be represented with $\{\vec{op} \times b\vec{b^{ij}}\}_{j=1}^{6}$, where $\vec{b^{ij}}$ denotes the jth direction vector of the bounding box i. Invalid intersections can be filtered out by checking if they are inside the 3D box. Finally, only the closest intersection j^* to the camera is selected when occlusion occurs:

$$j^* = \arg\min_{j} \{\vec{op} \times b\vec{b^{ij}}|_z\}_{j=1}^{6}, \tag{2}$$

where $|_z$ denotes the depth in of the intersection. Figure 3.b provides an example for the occlusion. Since the direction vector is based on the estimated 3D box, the gradient from the photometric loss can be directly back-propagated to the estimated 3D boxes. We present the details of generating the direction vector in the Appendix.

Surface-to-plane offset. As visualized in Figure 3.c, most of the pixels satisfy the cube-shaped assumption, especially for the side and bottom parts of the car. However, we found that this assumption does not always hold for variant regions (*e.g.,* car's corners, windshield, etc.). This observation motivates us to design a regression head to model the object shape. Specifically, the regression head leans an offset ΔZ that fills the gap between the depth computed from the cube-shaped assumption with the actual depth. See an offset example in Figure 3.c. Through modeling this offset, the projection process is robust to variant shapes, leading to accurate supervision signals for object localization. Finally, given a point u_s in the 2D coordinate of the source image, its corresponding 3D point can be acquired by $p_s = \pi(b, u, K, \varDelta Z)$, where b, u, K, $\varDelta Z$ represent the 3D box, 2D point, camera intrinsic, and the estimated offset, respectively. Note that the box b can be selected from the ground truth and the estimated bounding boxes. In the labeled data, we adopt the ground truth boxes to let the network learn the offset. In the unlabeled data, we adopt the estimated 3D boxes and jointly optimize the box and offset.

4.2.2 Object-level Photometric Loss

After the view projection, we can acquire the pixels in the source view with $I_s(p_s, K)$ and $I_t(T_{s\to t}p_s, K)$, and compute their photometric consistency loss.

It should be noted that in practice, some pixels in the objects are less informative, and matching them makes learning unstable. We propose an uncertainty-aware model to deal with this problem. Specifically, we model the uncertainty of the surface-to-plane offset and treat it as a re-weighting factor when computing the object-level photometric loss. In particular, we model the distribution of the offset as a Laplacian distribution based on ℓ_1 error [16, 29]. And the loss for one object is represented as follows:

$$\mathcal{L}_{photo}(p_s, \hat{p}_{s\to t}) = \frac{1}{N_p}\sum_{i=1}^{N_p} \frac{\sqrt{2}}{\sigma_i} \| I_s(p_s^i, K), I_t([p_s^i, 1]T_{s\to t}^T, K)\| + \log \sigma_i, \tag{3}$$

where N_p is the number of points and σ_i is the standard deviation of the offset. Intuitively, this uncertainty reweighting is similar to the curriculum learning in pseudo labeling for object classification: iteratively enlarges the training set from easy to complex data.

4.3 Overall Loss

The overall loss function in our semi-supervised training framework is represented as follows:

$$\mathcal{L}_{sup} = \mathcal{L}_{det} + \lambda_1 \cdot \mathcal{L}_{output} + \lambda_2 \cdot \mathcal{L}_{photo}, \tag{4}$$

where $\mathcal{L}_{det}$ is from the detection loss with ground truth bounding boxes, λ_1 and λ_2 are the manually tuned hyper-parameters.

5 Experiments

To validate the effectiveness of our semi-supervised framework, we conduct experiments on the KITTI [13] and nuScenes [4] datasets.

Table 1. Experimental results of 3D detection accuracy (Car) with different numbers of labeled data on the KITTI validation set. The metrics of $AP|_{R40}$ with IoU threshold=0.7 on three difficulties (easy, moderate and hard) are reported. We randomly sample 10%, 50%, and 100% data from the KITTI training set as the labeled split and select all the data in "Eigen Clean" as the unlabeled split. The models are trained with different types of multi-view data and evaluated with a single image.

Multi-view	Method	10%			50%			100%		
		Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard
-	Baseline	10.13	7.25	6.24	18.52	14.56	12.53	21.99	16.32	14.48
Lidar	Multi-task	**13.89**	8.86	7.53	20.91	15.70	13.58	23.98	18.01	15.33
Stereo	Multi-task	12.56	8.93	5.45	20.12	15.14	12.48	23.21	17.21	15.03
	MVC-MonoDet	13.34	**9.14**	**7.75**	**21.52**	**16.40**	**14.83**	**26.85**	**18.63**	**15.37**
Video	Multi-task	10.49	6.87	5.15	19.14	14.67	12.56	22.36	16.71	14.56
	MVC-MonoDet	12.13	7.96	7.02	21.15	16.01	13.37	24.45	17.34	15.15

5.1 Datasets

In this section, we first introduce the datasets we used and then descirbe the related evaluation metrics for 3D object detection.

KITTI is a popular dataset to benchmark multiple autonomous driving tasks. The 3D detection split consists of 14,999 annotated key frames with 7,481 for training and 7,518 for testing. Each key frame contains calibrated images from the left and right cameras with annotated 3D bounding boxes. Each labeled key frame is also accompanied with three adjacent unlabeled frames for providing temporal information. For a fair comparison, we follow recent work [7, 6] and split the training set into training and validation subsets with 3,712 and 3,769 frames, respectively. For the unlabeled split, we follow the recent pseudo lidar based approach [37] and adopt the "Eigen clean" subset that does not have overlap with the detection validation set. The "Eigen clean" subset selects 14,490 unlabeled video frames from 45,200 frames in the "Eigen" set.

nuScenes is a large scale autonomous driving dataset, which contains 1,000 video sequences. The official protocol splits the video sequences into 700 for the training subset, 150 for the validation subset, and 150 for the test subset. nuScenes annotated the 3D bounding box on each key frames with up to annotated 40k images from 6 cameras. We utilize the annotated key frames as the labeled split for supervised training and the other frames as the unlabeled split for semi-supervised training.

Evaluation metrics For the KITTI dataset, we adopt the official $AP|_{R40}$ metric that averages the precision number over 40 recall points. The IoU threshold is set as 0.7 for "Car" and 0.5 for both "Pedestrian" and "Cyclist", respectively. Following the benchmark [13], we classify the instances into three kinds of difficulty (easy, moderate, and hard) based on their 2D bounding box height, the occlusion and truncation levels. For the nuScenes dataset, we adopt the official AP (average precision with threshold of 0.5m, 1.0m, 2m, and 4m) and ATE

Table 2. Experimental results of Pedestrian and Cyclist on the KITTI validation set. (100% of labeled data is used.) The metrics of $AP|_{R40}$ with IoU threshold=0.7 on three difficulties (easy, moderate and hard) are reported.

Method	Pedestrian			Cyclist		
	Easy	Moderate	Hard	Easy	Moderate	Hard
Baseline	7.02	5.53	5.86	5.37	2.95	2.87
Multi-task	**8.44**	**6.89**	5.83	6.13	**4.10**	**3.96**
MVC-MonoDet	8.04	6.26	**6.94**	**6.94**	4.04	3.94

(average translation error) to evaluate the localization accuracy of the trained detectors.

5.2 Experimental setup

For a fair comparison, we initialize the network backbone (a modified version of DLA-34) with ImageNet [10] pre-trained weights and optimize the network by AdamW optimizer. The learning rate is set as 3e-4 and 1e-4 on the KITTI and nuScenes datasets, respectively. To select the foreground pixels for computing the photometric loss, we adopt a pre-trained segmentation model [46] to filter out the background pixel. In the semi-supervised training stage, we first pre-train the detector on the labeled subset with 70 epochs and 10 epochs for the KITTI and nuScenes dataset, respectively. Then we fine-tune the detector with the proposed semi-supervised framework on both the labeled and unlabeled subsets with extra 70 epochs for the KITTI dataset and 10 epochs for the nuScenes dataset. We set the training batch size as 8 and train the model on one NVIDIA 2080Ti GPU. Regarding the input data, we pad the images to the size of 1280×384 on the KITTI dataset and downsample the images to half of the resolution (800 × 450) on the nuScenes dataset. During inference, only a single image is fed to the detector and the image resolution is kept as in training.

For the ego pose, we directly adopt the calibrated ego pose provided in the dataset for training. In the video framework, we utilize the calibrated ego pose provided in the dataset for training. If the ego motion is unavailable, one also can adopt a motion network to estimate the object motion. To tackle dynamic objects in the video data, we follow [9, 18] previous work that models the corresponding object motion across frames. We repeat the experiments with three different random seeds and record their average value on the validation set.

5.3 Experimental results on the KITTI validation set

In Table 1, we represent the experimental results of our framework and the other competitors on the KITTI validation set. We conduct experiments with different numbers of labeled data, including 10%, 50%, and 100% of data sampled from the training subset. To the best of our knowledge, there is no other semi-supervised monocular 3D object detection approach sharing the same setting with us. Hence, except for fully supervised training and our approach, we

Table 3. Experimental results of 3D detection accuracy ($AP|_{R40}$ with IoU threshold =0.7) on the KITTI test benchmark. The best and second best results are marked with **bold** and blue color, respectively. "-" denotes that the method does not report the related statistics. EC denotes the clean subset of Eigen split. DDAD denotes the 15M private driving dataset in [30].

Setting	Extra	Method	Easy	Moderate	Hard	FPS
Vanilla	MonoFlex [50]	None	19.94	13.89	12.07	30
	Mono R-CNN [36]	None	18.36	12.65	10.03	70
	AutoShape [24]	None	22.47	14.17	11.36	50
	MonoRun [5]	None	19.65	12.30	10.58	70
	M3DSSD [26]	None	17.51	11.46	8.98	-
	Kinemantic [3]	None	19.07	12.72	9.17	-
	MonoDLE [29]	None	17.23	12.26	10.29	40
	GUP-Net [25]	None	20.11	14.20	11.77	30
	MonoEF [55]	Eigen	21.29	13.87	11.71	30
Pseudo-lidar	PatchNet [27]	Eigen	15.68	11.12	10.17	488
	PCT [43]	Eigen	21.00	13.37	11.31	487
	Demystifying [37]	EC	22.40	12.53	10.64	488
Multi-task	DD3D [30]	EC+DDAD	23.22	13.64	14.20	148
Direct-based	Baseline	None	20.63	13.21	11.05	30
	MVC-MonoDet	EC	**25.05**	**16.89**	**14.83**	30

further provide a "multi-task" framework [30] for comparison. The multi-task framework adds a parallel head on the modified CenterNet for depth estimation. In the multi-task framework, the supervision signal of depth estimation from either lidar, stereo or video can be used to update the joint feature representation. When using the stereo unlabeled data, our method outperforms the baseline approach with different numbers of labeled data, with ratios of 31.63%, 16.20%, and 22.10% on the easy split for three kinds of settings, respectively. The improvements validate the effectiveness of our approach in leveraging unlabeled data to improve the performance of the baseline. Given 50% labeled data, our approach even achieves comparable results with the baseline module that uses 100% labeled data.

When the number of labeled data is scarce, our approach still can improve the baseline module and the multi-task method. We also observe that the improvement is limited in the scarce data, and the potential reason is that the consistency modules need few labeled data to control the training. Furthermore, the large margin improvement on the 50% and 100% of labeled data also demonstrates the effectiveness of our approach providing direct supervision signals on the estimated bounding boxes. Compared between different modalities, the video version is not as effective as the stereo version, but still yields consistent improvements over the baseline approach. The performance gap between the stereo and video versions may come from the noise of ego and object motions in the video data.

To evaluate the effectiveness of our proposed approach in the non-rigid class, we also display the results of the pedestrian and cyclist classes in Table 2. Although the improvement is less ineffective than the car class, our approach yields consistent improvements over the baseline and multi-task framework, illustrating the flexibility of our framework in handling different kinds of objects. Note that the number of annotated instances in the pedestrian and cyclist is small (Pedestrian: 4,487, Cyclist: 1,627, and Car: 28,742). This may introduce performance fluctuations.

Table 4. Experimental results of different numbers of training data on the nuScenes validation set.

Setting	10%		100%	
	mAP↑	ATE↓	mAP↑	ATE↓
Baseline	15.9	0.87	33.2	0.68
Multi-task	17.2	0.86	33.6	0.67
MVC-MonoDet	18.8	0.82	34.9	0.64

5.4 Comparison with state-of-the-art detectors on the KITTI test set

Table 3 displays the comparison between our approach with state-of-the-art monocular detection methods on the KITTI test set. As illustrated, our framework outperforms the fully supervised detectors by a large margin and against the second-best approach with 14.02%, 14.29%, and 20.22% on the “Easy”, “Moderate” and “Hard” settings.

Compared to the pseudo-lidar based approaches, our method uses cheaper and less training data, in which pseudo-lidar based approaches utilize the full Eigen set (23,488) with lidar sensor while we use the Eigen clean subset (14,490). Alhtough using less training data, our approach still achieves much better performance. Compared with the multi-task framework, our framework does not utilize the extra private pre-trained dataset but still achieves better performance. This also demonstrates the effectiveness of the designed direct-based module for improving detection. Regarding the runtime efficiency, we follow previous work [50, 55] and evaluate the frame per second (FPS) on the RTX-2080Ti. Benefits from the semi-supervised training framework, the detector is improved and keeps high efficiency.

5.5 Experimental results on the nuScenes dataset

Except for the KITTI dataset, we also provide the experimental results of MVC-MonoDet on the nuScenes dataset. Since the nuScenes dataset only uses the monocular camera to collect image data, we provide the experimental results with our video framework. Table 4 displays the results of detectors trained with

10% and 100% of labeled data in the training subset. Similar to the observation on the KITTI dataset, our approach consistently improves the fully-supervised baseline and multi-task semi-supervised framework in both the mAP and ATE metrics.

5.6 Ablation study

In Table 5, we present the ablation study for different consistency regularization modules in our semi-supervised training framework. The ablation study is conducted on the KITTI dataset and 100% of labeled data is used in semi-supervised training. As shown in Table 5, both the box-level and object-level can effectively improve the baseline method. Meanwhile, these two kinds of regularization improve the detector from a different perspective, box-level is through enhancing model robustness and object-level is by latent appearance-based localization supervision. As a result, their combination can better improve the baseline method in different multi-view data.

Table 5. Ablation study of different components in our MVC-MonoDet framework. The mAP of Car category on the KITTI validation set is reported.

Multi-view	Box-level	Object-level	Easy	Moderate	Hard
-	-	-	21.99	16.32	14.48
Stereo	✓		24.93	17.56	14.71
		✓	24.16	17.85	14.92
	✓	✓	**26.85**	**18.63**	**15.37**
Video	✓		23.14	17.01	15.02
		✓	23.21	16.75	14.71
	✓	✓	**24.45**	**17.34**	**15.15**

6 Conclusion

In this paper, we proposed a semi-supervised monocular 3D object detection framework that leverages the unlabeled multi-view data (stereo or video) to improve performance. In the framework, we provide a box-level and an object-level consistency regularization to improve the performance of 3D detection. The box-level regularization provides sparse supervision to enhance the model's cross-view generalization on the unlabeled and unseen data. The object-level regularization utilizes dense supervision to explicitly identify and mitigate the bounding box prediction error. We showed that the designed regularization modules are effective in different types of multi-view data, leading to superior improvement over state-of-the-art results on the KITTI and nuScenes datasets.

References

1. Bian, J.W., Zhan, H., Wang, N., Li, Z., Zhang, L., Shen, C., Cheng, M.M., Reid, I.: Unsupervised scale-consistent depth learning from video. IJCV (2021)
2. Brazil, G., Liu, X.: M3d-rpn: Monocular 3d region proposal network for object detection. In: ICCV (2019)
3. Brazil, G., Pons-Moll, G., Liu, X., Schiele, B.: Kinematic 3d object detection in monocular video. In: ECCV (2020)
4. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: CVPR (2020)
5. Chen, H., Huang, Y., Tian, W., Gao, Z., Xiong, L.: Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation. In: CVPR (2021)
6. Chen, X., Kundu, K., Zhang, Z., Ma, H., Fidler, S., Urtasun, R.: Monocular 3d object detection for autonomous driving. In: CVPR (2016)
7. Chen, X., Kundu, K., Zhu, Y., Berneshawi, A.G., Ma, H., Fidler, S., Urtasun, R.: 3d object proposals for accurate object class detection. In: NeurIPS (2015)
8. Chen, Y., Tai, L., Sun, K., Li, M.: Monopair: Monocular 3d object detection using pairwise spatial relationships. In: CVPR (2020)
9. Dai, Q., Patil, V., Hecker, S., Dai, D., Van Gool, L., Schindler, K.: Self-supervised object motion and depth estimation from video. In: CVPRW (2020)
10. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
11. Dijk, T.v., Croon, G.d.: How do neural networks see depth in single images? In: ICCV (2019)
12. Ding, M., Huo, Y., Yi, H., Wang, Z., Shi, J., Lu, Z., Luo, P.: Learning depth-guided convolutions for monocular 3d object detection. In: CVPR (2020)
13. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)
14. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J.: Digging into self-supervised monocular depth prediction. In: ICCV (2019)
15. Jeong, J., Lee, S., Kim, J., Kwak, N.: Consistency-based semi-supervised learning for object detection. In: NeurIPS (2019)
16. Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? In: NeurIPS. Curran Associates, Inc. (2017)
17. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: Pointpillars: Fast encoders for object detection from point clouds. In: CVPR (2019)
18. Li, H., Gordon, A., Zhao, H., Casser, V., Angelova, A.: Unsupervised monocular depth learning in dynamic scenes. arXiv preprint arXiv:2010.16404 (2020)
19. Li, P., Chen, X., Shen, S.: Stereo r-cnn based 3d object detection for autonomous driving. In: CVPR (2019)
20. Li, P., Zhao, H., Liu, P., Cao, F.: Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. In: ECCV (2020)
21. Li, P., Zhao, H., Liu, P., Cao, F.: Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. arXiv preprint arXiv:2001.03343 (2020)
22. Lian, Q., Ye, B., Xu, R., Yao, W., Zhang, T.: Geometry-aware data augmentation for monocular 3d object detection. arXiv preprint arXiv:2104.05858 (2021)
23. Liu, Y.C., Ma, C.Y., He, Z., Kuo, C.W., Chen, K., Zhang, P., Wu, B., Kira, Z., Vajda, P.: Unbiased teacher for semi-supervised object detection. In: ICLR (2021)

24. Liu, Z., Zhou, D., Lu, F., Fang, J., Zhang, L.: Autoshape: Real-time shape-aware monocular 3d object detection. In: ICCV (2021)
25. Lu, Y., Ma, X., Yang, L., Zhang, T., Liu, Y., Chu, Q., Yan, J., Ouyang, W.: Geometry uncertainty projection network for monocular 3d object detection. In: ICCV (2021)
26. Luo, S., Dai, H., Shao, L., Ding, Y.: M3dssd: Monocular 3d single stage object detector. In: CVPR (2021)
27. Ma, X., Liu, S., Xia, Z., Zhang, H., Zeng, X., Ouyang, W.: Rethinking pseudo-lidar representation. In: ECCV (2020)
28. Ma, X., Wang, Z., Li, H., Zhang, P., Ouyang, W., Fan, X.: Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving. In: ICCV (2019)
29. Ma, X., Zhang, Y., Xu, D., Zhou, D., Yi, S., Li, H., Ouyang, W.: Delving into localization errors for monocular 3d object detection. In: CVPR (2021)
30. Park, D., Ambrus, R., Guizilini, V., Li, J., Gaidon, A.: Is pseudo-lidar needed for monocular 3d object detection? In: ICCV (2021)
31. Qian, S., Sun, K., Wu, W., Qian, C., Jia, J.: Aggregation via separation: Boosting facial landmark detector with semi-supervised style translation. In: ICCV (2019)
32. Reading, C., Harakeh, A., Chae, J., Waslander, S.L.: Categorical depth distribution network for monocular 3d object detection. In: CVPR (2021)
33. Shi, S., Guo, C., Jiang, L., Wang, Z., Shi, J., Wang, X., Li, H.: Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In: CVPR (2020)
34. Shi, S., Wang, X., Li, H.: Pointrcnn: 3d object proposal generation and detection from point cloud. In: CVPR (2019)
35. Shi, X., Chen, Z., Kim, T.K.: Distance-normalized unified representation for monocular 3d object detection. In: ECCV (2020)
36. Shi, X., Ye, Q., Chen, X., Chen, C., Chen, Z., Kim, T.K.: Geometry-based distance decomposition for monocular 3d object detection. In: ICCV (2021)
37. Simonelli, A., Bulò, S.R., Porzi, L., Kontschieder, P., Ricci, E.: Are we missing confidence in pseudo-lidar methods for monocular 3d object detection? In: ICCV (2021)
38. Simonelli, A., Bulò, S.R.R., Porzi, L., López-Antequera, M., Kontschieder, P.: Disentangling monocular 3d object detection. arXiv preprint arXiv:1905.12365 (2019)
39. Sohn, K., Zhang, Z., Li, C.L., Zhang, H., Lee, C.Y., Pfister, T.: A simple semi-supervised learning framework for object detection. arXiv preprint arXiv:2005.04757 (2020)
40. Tang, P., Ramaiah, C., Wang, Y., Xu, R., Xiong, C.: Proposal learning for semi-supervised object detection. In: WACV (2021)
41. Wang, H., Cong, Y., Litany, O., Gao, Y., Guibas, L.J.: 3dioumatch: Leveraging iou prediction for semi-supervised 3d object detection. In: CVPR (2021)
42. Wang, L., Du, L., Ye, X., Fu, Y., Guo, G., Xue, X., Feng, J., Zhang, L.: Depth-conditioned dynamic message propagation for monocular 3d object detection. In: CVPR (2021)
43. Wang, L., Zhang, L., Zhu, Y., Zhang, Z., He, T., Li, M., Xue, X.: Progressive coordinate transforms for monocular 3d object detection. In: NeurIPS (2021)
44. Wang, Y., Chao, W.L., Garg, D., Hariharan, B., Campbell, M., Weinberger, K.Q.: Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In: CVPR (2019)
45. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. TIP **13**(4), 600–612 (2004)

46. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. https://github.com/facebookresearch/detectron2 (2019)
47. Xie, J., Girshick, R., Farhadi, A.: Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks. In: ECCV (2016)
48. You, Y., Wang, Y., Chao, W.L., Garg, D., Pleiss, G., Hariharan, B., Campbell, M., Weinberger, K.Q.: Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving. arXiv preprint arXiv:1906.06310 (2019)
49. Zhang, F., Pan, T., Wang, B.: Semi-supervised object detection with adaptive class-rebalancing self-training. arXiv preprint arXiv:2107.05031 (2021)
50. Zhang, Y., Lu, J., Zhou, J.: Objects are different: Flexible monocular 3d object detection. In: CVPR (2021)
51. Zhao, N., Chua, T.S., Lee, G.H.: Sess: Self-ensembling semi-supervised 3d object detection. In: CVPR (2020)
52. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised learning of depth and ego-motion from video. In: CVPR (2017)
53. Zhou, X., Koltun, V., Krähenbühl, P.: Tracking objects as points. In: ECCV (2020)
54. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. In: arXiv preprint arXiv:1904.07850 (2019)
55. Zhou, Y., He, Y., Zhu, H., Wang, C., Li, H., Jiang, Q.: Monocular 3d object detection: An extrinsic parameter free approach. In: CVPR (2021)