PseudoAugment: Learning to Use Unlabeled Data for Data Augmentation in Point Clouds

Zhaoqi Leng[1*], Shuyang Cheng[1], Benjamin Caine[2], Weiyue Wang[1], Xiao Zhang[1], Jonathon Shlens[2], Mingxing Tan[1], and Dragomir Anguelov[1]

Waymo[1], Google[2]
lengzhaoqi@waymo.com*

Abstract. Data augmentation is an important technique to improve data efficiency and to save labeling cost for 3D detection in point clouds. Yet, existing augmentation policies have so far been designed to only utilize labeled data, which limits the data diversity. In this paper, we recognize that pseudo labeling and data augmentation are complementary, thus propose to leverage unlabeled data for data augmentation to enrich the training data. In particular, we design three novel pseudo-label based data augmentation policies (PseudoAugments) to fuse both labeled and pseudo-labeled scenes, including frames (PseudoFrame), objects (PseudoBBox), and background (PseudoBackground). PseudoAugments outperforms pseudo labeling by mitigating pseudo labeling errors and generating diverse fused training scenes. We demonstrate PseudoAugments generalize across point-based and voxel-based architectures, different model capacity and both KITTI and Waymo Open Dataset. To alleviate the cost of hyperparameter tuning and iterative pseudo labeling, we develop a population-based data augmentation framework for 3D detection, named AutoPseudoAugment. Unlike previous works that perform pseudo-labeling offline, our framework performs PseudoAugments and hyperparameter tuning in one shot to reduce computational cost. Experimental results on the large-scale Waymo Open Dataset show our method outperforms state-of-the-art auto data augmentation method (PPBA) and self-training method (pseudo labeling). In particular, AutoPseudoAugment is about 3× and 2× data efficient on vehicle and pedestrian tasks compared to prior arts. Notably, AutoPseudoAugment nearly matches the full dataset training results, with just 10% of the labeled run segments on the vehicle detection task.

1 Introduction

3D object detection from LiDAR point cloud data is a core component of autonomous driving. Building accurate 3D object detection systems requires vast quantities of labeled scenes with accurate 3D bounding box annotations. While unlabeled LiDAR data is readily available, labeling itself is costly, e.g., 6.4 hours of LiDAR data contains more than 10 million human labeled 3D boxes [58]. Because of this, an effective way to increase the data efficiency for model training would be very appealing.

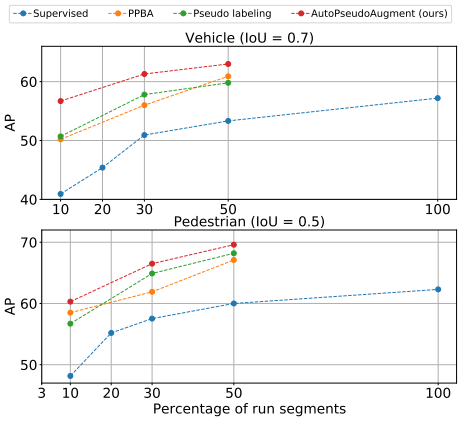

Fig. 1: **AutoPseudoAugment is more data efficient than auto data augmentation and self-training methods.** Data augmentation only (PPBA [8]), self-training (Pseudo labeling [4]) and our method (AutoPseudoAugment) are evaluated using 3D detection AP at Level 1 difficulty on the *validation* split of the Waymo Open Dataset [58]. Using 10% of labeled run segments, AutoPseudoAugment is about 3× data efficient as PPBA and Pseudo label method on the vehicle class and 2× on the pedestrian class. AutoPseudoAugment is nearly 10× and more than 5× data efficient compared to the supervised (no augmentation) vehicle and pedestrian baselines.

Data augmentation represents an effective way to increase data efficiency for labeled data. Data augmentations for 3D detection generally come in two forms: global augmentations like scene rotations, or local augmentations like ground truth augmentation, where crops of ground truth objects from the training set are inserted into the scene. Pasting ground truth objects into the scene has been shown to be extremely effective on various 3D detection datasets [64,25,8,66,70,65,30].

However, these augmentation techniques are typically limited to the labeled training data. A simple way to incorporate unlabeled data into training is pseudo labeling, but naively applying existing 3D data augmentation policies to pseudo labeled frames has an intrinsic limitation, i.e., pseudo labeled frames contain numerous false positive/negative bounding boxes and points. Several recent studies on 3D pseudo labeling [4,44] have tried to use large-capacity teacher models to mitigate this issue, but the intrinsic pseudo-labeling errors persist. Here, we seek to an alternative approach: *mitigating the pseudo labeled errors by new data augmentation policies.*

Another challenge is how to effectively combine labeled and unlabeled data via data augmentation. Previous approaches treat pseudo-labeled frames as a whole and do not recognize the compositional nature of 3D point clouds scenes [4,44]. This limits the diversity of training data. A simple way to fuse labeled and pseudo-labeled frame is to generalize the existing copy-pasting object

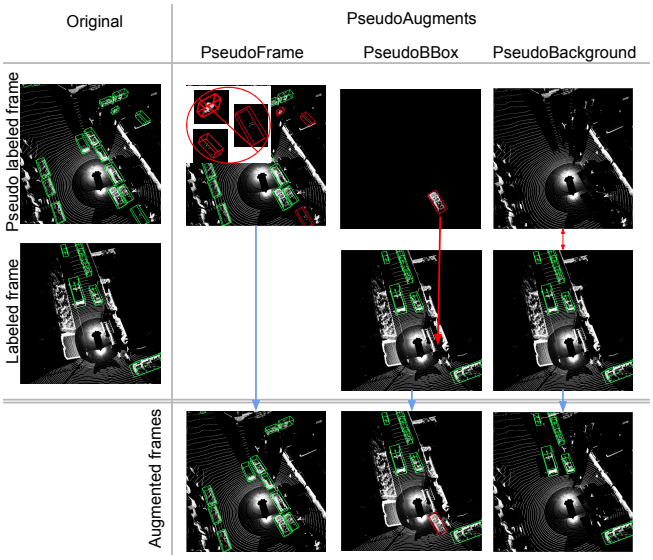

Fig. 2: **Visualization of PseudoAugments**. PseudoAugments contain three new data augmentation policies: PseudoFrame, PseudoBBox, and PseudoBackground. **PseudoFrame** replaces the labeled frame with a pseudo-labeled frame and drops points of low-confidence bounding boxes in the pseudo-labeled frames. **PseudoBBox** pastes high-confidence bounding boxes and corresponding point clouds from a pseudo-labeled frame to a labeled frame. **PseudoBackground** removes all points within bounding boxes in a pseudo-labeled frame, and replaces the background point clouds in the labeled frame with the background point clouds of the pseudo-labeled frame. The augmented frames are used as labeled frames during training.

data augmentation to leverage unlabeled objects. Interestingly, we observe that only pasting objects between labeled and pseudo-labeled frames is not enough [64,25,8,66,70,65,30], because we miss out the diverse background scenes in the pseudo labeled dataset. Especially for 3D point clouds, more than 90% of the points are backgrounds, which provide critical ingredients for 3D detectors to learn to detect objects in new scenarios. Thus, it is necessary to develop a set of data augmentation policies that *take advantage of both foreground objects and background points in the pseudo labeled frames along with labeled frames to generate combinatorial number of point clouds.*

In this work, we propose a set of data augmentation policies tailored for pseudo labeled data, named *PseudoAugments*. As shown in Figure 2, our PseudoAugments contain three new data augmentation policies: *PseudoFrame* removes low confidence points, *PseudoBBox* pastes pseudo objects onto labeled scenes, and *PseudoBackground* swaps the background point clouds between labeled and pseudo-labeled scenes. All our augmentations allow pseudo-labeling

uncertainty, and only make use of points of frame, object, and background with high-confidence. The three new PseudoAugments significantly increase the diversity of training data by enabling a combinatorial number of new *fused* training scenes, including 1) ground truth objects on pseudo labeled background scenes, 2) pseudo labeled objects on ground truth background scenes, and 3) pseudo labeled objects on pseudo labeled background scenes, which greatly enrich the diversity of training data.

Based on PseudoAugments, we develop an auto data augmentation framework named AutoPseudoAugment to learn the best augmentation policies. Our AutoPseudoAugment is based on population-based training (PBT) and online search for the best augmentation policies at different training stages. On top of PBT, AutoPseudoAugment also uses the top-performing models in previous generations as an ensemble of teachers to pseudo label unlabeled data, which further boost the quality of pseudo labeled data without the need of training a separated set of high-capacity teacher models [4,44]. AutoPseudoAugment also extends PBT beyond simple hyperparameter tuning by introducing population-based distillation and creates a virtuous cycle between students and teachers, where good teachers in previous generations improves the quality of student models, which become better teachers to pseudo label for future generations.

Our main contribution can be summarized as follows:

1. **PseudoAugments: unifying data augmentation and pseudo labeling.** We identify data augmentation and pseudo labeling are complementary and introduce PseudoFrame, PseudoBBox, PseudoBackground data augmentation policies to take advantage of the composability of unlabeled 3D point clouds while mitigating errors.
2. **AutoPseudoAugment: efficient one-shot framework for PseudoAugment**. Our framework extends PBT by introducing population-based distillation. AutoPseudoAugment does auto hyperparameters search and self-training in one-shot, which reduces the training cost.
3. **Extensive experimental evaluations.** We demonstrate PseudoAugments generalize to different network architectures, model sizes, and datasets. In addition, AutoPseudoAugment outperforms both state-of-the-art auto data augmentation method (PPBA [8]) and pseudo labeling [4]. In particular, leveraging unlabeled data, AutoPseudoAugment requires 10% of labeled run segments to achieve similar performance as PPBA training on 30% of run segments and nearly matches the model performance trained on all labeled data without data augmentation, shown in Figure 1.

2 Related Work

2.1 Data augmentation

Data augmentation has been widely adopted to improve the performance of models trained with supervised learning, such as image classification[55,10,60,49,29,13,67], 2D object detection[23,14], image segmentation[46,38,48,15], point cloud classification and detection[70,64,7,30,39,51,34,8,18,50,28,31,9,68].

Due to the number of hyperparameters introduced by using a suite of data augmentations during training, designing a strong augmentation policy for a given task and dataset requires extensive experimentation. Automated data augmentation algorithms [33,45,11,12,35,24] were proposed to search data augmentation policies. Recently, PointAugment [34] and PPBA [8] introduced automated data augmentation for point clouds, which showed strong empirical results.

Unlike existing data augmentation methods, which only operates on labeled data, our PseudoAugments are designed to improve the quality of pseudo labeled data and generate combinatorially diverse scenes by fusing labeled and pseudo labeled frames. Different from automated data augmentation frameworks, in particular population-based data augmentation [24,8], our AutoPseudoAugment framework enables hyperparameters tuning and self-training in one-shot. It reduces the training cost especially for iterative self-training [63] and outperformed the state-of-the-art data augmentation framework for 3D point clouds, shown in Table 4.

2.2 Self-training

Self-training [36,63,6,4], also referred to as pseudo-labeling [32], aims to learn from a combination of labeled and unlabeled data. In self-training, a trained teacher network is used to predict labels (pseudo labels) on unlabeled data, and a student model is later trained on the combination of the original labeled examples and the new pseudo-labeled examples. Self-training has been applied to a wide variety of tasks, including classification [63,56,1], semantic segmentation [40,6,62,22], object detection [47,57,71,4], speech recognition [41,27].

Different from prior works on pseudo labeling for 3D point cloud [4,44,61], where unlabeled frames are used as a whole, our PseudoAugments enables combinatorial new training data by fusing labeled and unlabeled frames. In this work, we aim to demonstrate simple PseudoAugment policies are effective and general methods, while advanced techniques such as IoU-based filtering [61], part&shape-aware data augmentation [68,9], and randering-based method [18] could further improve the quality of PseudoAugments.

2.3 Object Detection for Point Clouds

There exists a large collection of different architectures for performing 3D Object Detection. The majority of methods [7,66,30,70,64,19,16] discretize the space into either a 2D (Birds eye view) or 3D grid, and perform either 2D or 3D convolutions on this grid. Some methods alternatively opt to work with the range image view, performing convolutions on the spherical LiDAR image [37,2,17]. There also exists a third class of methods, that opt to learn features directly from the raw point cloud [43,39,54,42], along with a handful of techniques that blend approaches [69,52,53,59]. Because our method is architecture-agnostic, we view these innovations as complimentary, as our method should benefit current and future architectures.

3 Methods

In this section, we first motivate PseudoAugment policies and explain their designs, then we detail the AutoPseudoAugment framework including our overall data augmentation search process and how this interacts with these PseudoAugment policies. A summary of the algorithm can be found in Algorithm 1.

Algorithm 1 AutoPseudoAugment contains two new elements: population-based distillation and PseudoAugments.

Input: data and label $(\mathcal{A}, \mathcal{B})$ and unlabeled data $\mathcal{C}$
Init: set training step $t = 0$, total training steps $\mathcal{N}$, generation step K, randomly initialize M models with random PseudoAugment hyperparameters θ.
while $t \neq \mathcal{N}$ **do**
  if $mod(t, K) == 0$ **then**
    # Population based distillation
    Select the top N models in the previous generation to pseudo label unlabeled data $\mathcal{C}$ and store into a pseudo database which contains unlabeled data and pseudo label $(\mathcal{C}, \mathcal{D})$
    # Standard progressive PBT exploitation and exploration
    Update hyperparameters θ and model parameters based on PBT [26,8]
  else
    # Model trained with PseudoAugment policies
    Independently train M models in parallel while using the pseudo database $(\mathcal{C}, \mathcal{D})$ to augment the training data $(\mathcal{A}, \mathcal{B})$ through PseudoAugment policies.
  end if
end while

3.1 PseudoAugments

The primary goal of PseudoAugments is to generate more diverse training data by fusing pseudo-labeled and labeled frames, while reducing misclassified points and objects in pseudo-labeled frame. We proposed three new data augmentation polices which corresponds three different ways of utilizing pseudo-labeled data: PseudoFrame, PseudoBBox, PseudoBackground.

PseudoFrame. PseudoFrame extends the self-training approach, where a pseudo-labeled frame is used as if a labeled frame during training. Unlike [4], where pseudo bounding boxes with low prediction confidence are dropped to suppress false positive bounding boxes, PseudoFrame augments pseudo-labeled frames by truncating point clouds within those low confidence pseudo bounding boxes, shown in Figure 2. In fact, pseudo labeling is suboptimal compared to PseudoFrame, regardless of what the confidence threshold is, e.g., setting a high threshold will introduce false negative points in the scenes while setting a low threshold will lead to additional false positive pseudo-labeled bounding boxes in the scenes. PseudoFrame resolves this challenge by simply dropping point clouds in confusing (low confidence) pseudo boxes, which increases the effective quality of pseudo-labeled data, Figure 4, and leads to higher quality student models, Table 1. The PseudoFrame policy is simple and contains only two hyperparameters: the probability of applying this policy (range [0, 1]), and the threshold

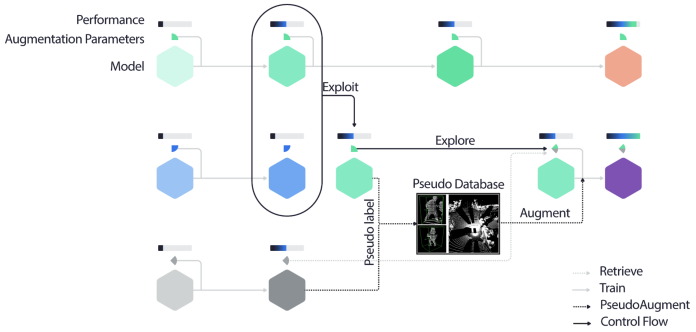

Fig. 3: **Schematic diagram of AutoPseudoAugment.** AutoPseudoAugment extends the idea of population-based data augmentation by introducing population-based distillation. Population-based distillation is done at the end of each generation, where we select top-performing models (green and grey hexagons) in previous generations as an ensemble of teachers to pseudo label unlabeled data. The pseudo-labeled frames are stored in a pseudo database, which are used to augment input point clouds in the next generation of training. In practice, we follow the recommendations of PPBA [8] and only explore up to three augmentation policy choices per generation, with exploration rate 0.8. More details on population based data augmentation can be found in [24,8].

of the classification confidence score for both dropping bounding boxes as well points (range [0.5, 1]).

Though PseudoFrame can leverage unlabeled data and increase the effective quality of pseudo-labeled data, the labeled frames and pseudo-labeled frames are still trained independently. To further increase the diversity of the training data, we introduce two more data augmentation policies, i.e., PseudoBBox and PseudoBackground, to fuse labeled frames and pseudo-labeled frames, which introduce combinartorial number of new training data.

PseudoBBox. Unlike pasting ground truth objects from the labeled frames [64,8,39,30], PseudoBBox is designed to introduce diverse pseudo objects into a training example while reducing the likelihood of pasting false positive points as objects, shown in Figure 2. PseudoBBox fuses pseudo-labeled frames and labeled frames by pasting pseudo-labeled foreground objects onto labeled scenes. The PseudoBBox policy contains three parameters: the probability of applying this policy (range [0, 1]), the number of objects that will be added (range [0, 20]), and the threshold of the classification confidence score (range [0.5, 1]) required for an object to be inserted into a scene.

To align pasted objects to the new background scene, we adjust the Z value based on an estimate of the ground plane's Z coordinate [1]. We oversample 10× pseudo objects and reject pseudo objects that overlap with any other pseudo objects and existing ground truth objects in the scene, then sample from the reminding pseudo objects and paste the predefined number of pseudo objects into the scene. If the pasted objects overlap with background points, we will remove background points.

PseudoBackground. Perhaps surprisingly, the background point clouds in unlabeled data contain important ingredients for generating diverse fused scenes, which are not recognized before. Simply swapping the background point clouds in labeled frames and unlabeled frames, we can generate diverse fused training scenes with ground truth objects on top of background point clouds from pseudo-labeled scenes. Different from PseudoFrame and PseudoBBox, we aggressively reject both true negative and false negative points in point clouds by removing all points within pseudo bounding boxes with object classification confidence scores above 0.1, and use reminding points as the background point clouds. Thus, the PseudoBackground is simple and contains only one hyperparameter, i.e., the probability of applying this operator (range [0, 1]). We align the ground plane of the new pseudo background point cloud with the existing point cloud and reject pseudo background point clouds when overlapping with bounding boxes, following the process described above for PseudoBoundingBox.

3.2 AutoPseudoAugment

AutoPseudoAugment is a data augmentation framework designed for efficient hyperparameter tuning while applying PseudoAugments in one shot, shown in Algorithm 1.

Population-based distillation Motivated by the recent success of population-based augmentation [24,8], we apply PBT to find the optimal hypperparameters in PseudoAugments. However, traditionally, hyperparameter tuning and self-training are decoupled. Especially for iterative self-training [63], tuning the hyperarameters for the student model in each iteration will incur significant computation cost. To mitigate this challenge, we propose population-based distillation, where we take advantage of the models in previous generations as an ensemble of teachers to pseudo label unlabeled data, shown in Figure 3.

Unlike PBT, where past generation model checkpoints are discarded when training the current generation, we recycle and use the top N model checkpoints in the previous generation as teachers. Because the previous generation checkpoints are trained with different schedules of data augmentation policies, they naturally form a diverse set of teachers. Thus, population-based distillation achieves both hyperparameter tuning and ensemble distillation at once.

In addition to our three new PseudoAugment policies PseudoFrame, PseudoBBox, and PseudoBackground, we adopt the full suite of data augmentations

[1] We estimate this with linear regression of the bottom center of the foreground ground truth or pseudo-labeled objects. If less than 3 bounding boxes are in the scene, we use the histogram of point clouds Z coordinate to estimate the ground plane.

used by PPBA [8]. In order to further increase the diversity of our training data, we apply our three PseudoAugment policies *before* we apply other geometric-based data augmentations, allowing pseudo-label augmented scenes to be further augmented by other common data augmentations. Our final order of augmentations that could potentially be applied (given the choices of the policy) are: PseudoFrame, PseudoBoundingBox, PseudoBackground, RandomRotation, WorldScaling, GlobalTranslateNoise, FrustumDropout, FrustumNoise, RandomDropLaserPoints.

In this section, we extensively evaluate PseudoAugments policies and AutoPseudoAugment framework using voxel-based PointPillars model [2] [30] and point-based StarNet model [2] [39] on KITTI [20] and Waymo Open Dataset [58].

For the following experiments, we train two separate models to detect vehicles and pedestrians and adopt the same training setting as prior works [39,8,4]. To study the data efficiency, we create a smaller training set consisting of 10%, 30% and 50% of the run segments from the Waymo Open Dataset training set to use as our labeled dataset, while using the remaining run segments as an unlabeled dataset. We want to highlight that 10% of the Waymo Open Dataset contains a considerable amount of 3D labeled bounding boxes (more than 1 million) which is on par with other full training dataset such as KITTI, NuScenes, and Argoverse dataset [21,3,5]. For hyperparameter tuning on Waymo Open Dataset, we create a random subsampling of the validation set, using 10% of examples (4109 samples) as *mini-val* and use Level 1 difficulty average precision (AP) as our objective value.

4 Experiments

4.1 PseudoAugments helps quality and diversity

In this section, we show PseudoAugments reduce the errors in pseudo labeled scenes via PseudoFrame and can generate diverse fused scenes when applying PseudoBBox and PseudoBackground, which outperform pseudo labeling method for both vehicle and pedestrian detection tasks. We follow the implementation in [4] and train teacher models on 10% of the training run segments using random Z rotation and random flip Y data augmentation for 150 epochs. We use the teacher models to pseudo label the reminding 90% of the training run segments and remove pseudo-labeled bounding boxes with classification score below 0.5. When training the student models, we use 1:1 ratio of labeled and pseudo labeled scene in each mini batch. Since the training data is increased 10×, we train the student model for 10× steps to take advantage of the additional pseudo labeled data, results shown in Table 1.

PseudoFrame improves data quality. PseudoFrame augments the pseudo labeled scenes by removing point clouds in not so confident pseudo bounding boxes. Here, we remove bounding boxes and corresponding point clouds with

[2] Code for both models are available at https://github.com/tensorflow/lingvo/tree/master/lingvo/tasks/car under Apache License 2.0.

Setup	Effects	Vehicle L1 AP	Pedestrian L1 AP
Supervised (Teacher)		49.6	53.9
Pseudo labeling [4]		50.7 (+1.1)	56.7 (+2.8)
PseudoFrame only (Ours)	Reducing errors	51.1 (+1.5)	57.2 (+3.3)
PseudoBBox only (Ours)	Fusing scenes	53.4 (+3.8)	57.0 (+3.1)
PseudoBackground only (Ours)	Fusing scenes	51.9 (+2.3)	57.7 (+3.8)
All PseudoAugments (Ours)	Reducing errors + fusing scenes	**54.3 (+4.7)**	**58.4 (+4.5)**

Table 1: **PseudoAugments improve upon Pseudo labeling method**. PseudoAugments reduce errors in the pseudo-labeled scenes by dropping low-confidence points, and improves data diversity by introducing fused pseudo-labeled scenes. Supervised PointPillars models are trained on 10% run segments and used as teachers. Pseudo labeling drops pseudo-labeled bounding boxes below confidence threshold 0.5, while PseudoFrame augments pseudo-labeled scenes by dropping both bouding boxes and point clouds within those bounding boxes below threshold 0.5. The improvements from PseudoAugments are additive. Introducing PseudoBBox and PseudoBackground further enrichs the training data, which leads to better student models. 3D detection Level 1 AP are evaluated on the Waymo Open Dataset *validation set*.

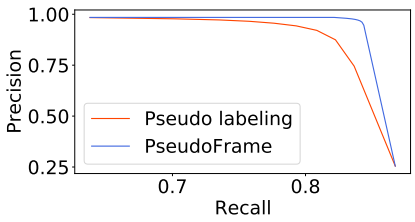

Fig. 4: **PseudoFrame improves quality of pseudo labeled point clouds.** Precision and recall are defined based on whether a point is inside labeled/pseudo labeled vehicle or pedestrian bounding boxes. Vanilla pseudo labeling approach only adds pseudo bounding boxes if the prediction confidence is higher than 0.5, but keeps all the false-negative points; In contrast, our PseudoFrame also drops points in low-confidence bounding boxes, thus reducing false negatives and improving precision-recall of pseudo labeled frames.

classification confidence score below 0.5. As shown in Figure 4, simply removing those point clouds is an effective data augmentation to increase the prevision-recall of pseudo labeled points. PseudoFrame improves the quality of student models (+0.4 on Vehicle AP and +0.5 on Pedestrian AP) compared to Pseudo labeling, shown in Table 1.

PseudoBBox and PseudoBackground increase diversity. PseudoBBox and PseudoBackground increase the diversity of the training scenes by fusing pseudo labeled and labeled scenes, as shown in Figure 2. To find the optimal hyperparameters, we randomly sample 16 different combinations of hyperparameters from the search space detailed in subsection 3.1. Introducing fused scenes

further improves the quality of student models (+3.2 on Vehicle AP and + 1.2 on Pedestrian AP) compared to only applying PseudoFrame data augmentation, Table 1, which shows the benefit of PseudoBBox and PseudoBackground is additive.

4.2 Generalization of PseudoAugments

In the previous section, we demonstrate PseudoAugments improves upon pseudo labeling method on PointPillars models. In this section, we show PsueodAugments generalizes to different model sizes and architectures. In addition to PointPillars model, which is a voxel-based architecture [30], we evaluate PseudoAugment on larger PointPillars models and point-based StarNet [39] models. We use the same pseudo labeled data as in subsection 4.1, which is labeled by the supervised PointPillars models shown in Table 1. We show besides self-training using the same model size and architectures, PseudoAugments enables self-training from a smaller model to a larger model and across different architectures. Our results show PseudoAugments lead to higher improvements compared to pseudo labeling, Table 2. For the following experiments, we adopt the same training settings as in subsection 3.1.

Setup	Vehicle AP	Pedestrian AP
Supervised	52.1	56.9
Pseudo labeling [4]	51.6 (-0.5)	57.8 (+0.9)
All PseudoAugments (Ours)	**55.7**(+5.5)	**59.7** (+2.8)

(a) Pillars2X.

Setup	Vehicle AP	Pedestrian AP
Supervised	43.7	60.6
Pseudo labeling [4]	48.2 (+4.5)	63.5 (+2.9)
All PseudoAugments (Ours)	**51.2** (+7.5)	**64.7** (+4.1)

(b) StarNet models.

Table 2: **PseudoAugments generalize to larger capacity models and different architectures.** PseudoAugments outperform pseudo labeling on 10% run segments using PointPillars [30], in Table 1, as teachers. (a) self-training from PointPillars teachers to larger PointPillars models (Pillars2X) and (b) self-training from PointPillars teachers to StarNet models [39]. Note that PseudoAugments improve the vehicle detection quality of Pillars2X whereas pseudo labeling is unable to. 3D detection Level 1 AP are evaluated on the Waymo Open Dataset *validation set*.

Generalize to larger models. We double the channel numbers of every convolution layers in the PointPillars model and denote it as Pillars2X. We train Pillars2X on the same supervised 10% run segments as the supervised training baseline, which has higher quality compared to the standard (1x) PointPillars, shown in Table 2a. Interestingly, the pseudo labeling method failed to improve the vehicle Pillars2X model when we use a weaker (1X) model as teacher (52.1 AP for supervised Pillars2X and 49.6 AP for supervised PointPillars on vehicle detection). This indicates errors in pseudo labeled frames diminishe the benefit of introducing unseen scenes to diversify the training data. Whereas, applying

PseudoAugments overcomes this limitation and leads to significant improvement (+4.1 on Vehicle AP and + 1.9 on Pedestrian AP) compared to pseudo labeling.

Generalize to different architectures. Unlike voxel-based PointPillars, StarNet is a point-based 3D detector and learns feature representations directly from raw point clouds. Our results show, using PointPillars model as teacher, PseudoAugments significantly improves quality of the StarNet student models (+3.0 on Vehicle AP and + 1.2 on Pedestrian AP) compared to pseudo labeling method Table 2b. This shows PseudoAugments are model agnostic and outperform pseudo labeling method when self-training between very different model architectures.

4.3 Generalize to KITTI dataset

In this section, we show PseudoAugments is a general method that is effetive on significantly different datasets. Different from Waymo Open Dataset [58], KITTI [20] dataset was collected in different cities and has different point and object density per frame. Here, we follow the common practice and split the KITTI dataset in half, i.e., one used for training and the other half used for validation. We randomly select 10% of the training frames as a mini training split, while removing labels on the rest 90% of the training frames. We train PointPillars teacher models on the mini training split with random flip and random world scaling data augmentations. Our results, in Table 3, show using PseudoAugments consistently outperform pseudo labeling on detecting objects at all difficulties.

Setup	Vehicle (E/M/H)	Ped&Cyc (E/M/H)
Supervised (Teacher)	55.6/49.2/46.1	46.3/33.4/30.2
Pseudo labeling [4]	64.3/51.4/49.0	49.3/35.9/32.6
All PseudoAugments (Ours)	**65.5/56.5/53.7**	**55.2/40.8/37.5**

Table 3: **PseudoAugments generalize to KITTI dataset**. PseudoAugments outperform pseudo labeling on 10% KITTI training frames using PointPillars [30] as teachers. 3D detection APs for easy, moderate, and hard (E/M/H) objects are evaluated on the KITTI *validation set*.

4.4 AutoPseudoAugment improves data efficiency

In previous sections, we demonstrate PseudoAugments are strong data augmentation methods that improves upon pseudo labeling. In this section, we show AutoPseudoAugment leverages PseudoAugments and further advances state-of-the-art auto data augmentation methods for 3D point clouds (PPBA) [8].

When the models are trained on 10% labeled run segments, we use generation step 1000 for both PPBA and AutoPseudoAugment. On 30% and 50% run segments, we increase the generation step to 2000. Even though AutoPseudoAugment introduces additional PseudoAugment policies compared to PPBA,

we use the same number of tuners (population size 16) for AutoPseudoAugment and PPBA. We follow the other training settings in [8]. At the end of each generation, we select the top 10 models in previous generations with L1 AP above 0.35 as ensemble of teachers to pseudo label unlabeled data.

AutoPseudoAugment outperforms both Pseudo labeling and PPBA methods Our AutoPseudoAugment framework subsumes both the auto data augmentation and pseudo labeling, which takes advantage of additional unlabeled data while tuning hyperparameters online. More importantly, PseudoAugments generate high-quality fused scenes, which greatly increases the diversity of the training data. As shown in Table 4, AutoPseudoAugment outperforms both PPBA and Pseudo labeling on 10%, 30%, and 50% labeled run segments.

To estimate the data efficiency, we train PointPillars models without data augmentation on 10%, 20%, 30%, 50% and 100% of training run segments, shown in Figure 1. According to this metric, our AutoPseudoAugment at 10% run segments (56.7 AP) is almost 10× more data efficient on the vehicle class, which nearly matches the model trained with 100% labeled data (57.2 AP). On pedestrian class, AutoPseudoAugment at 10 % run segments (60.3 AP) shows 5× data efficient and suprasses no augmentation baseline model trained on 50 % of the run segments (60.0 AP), shown in Figure 1.

			Vehicle					
Setup	Type of data	AutoML	10 %		30 %		50 %	
			AP (L1/L2)	APH (L1/L2)	AP (L1/L2)	APH (L1/L2)	AP (L1/L2)	APH (L1/L2)
PPBA [8]	Labeled only	✓	50.2/43.4	49.7/42.9	56.0/48.7	55.5/48.2	60.9/53.0	60.4/52.6
Pseudo labeling [4]	Labeled+Unlabeled		50.7/43.9	50.2/43.5	57.8/50.2	57.3/49.8	59.8/52.0	59.3/51.6
AutoPseudoAugment	Labeled+Unlabeled+Fused	✓	**56.7/49.2**	**56.3/48.8**	**61.3/53.5**	**60.9/53.1**	**63.0/55.1**	**62.5/54.6**
			Pedestrian					
Setup	Type of data	AutoML	10 %		30 %		50 %	
			AP (L1/L2)	APH (L1/L2)	AP (L1/L2)	APH (L1/L2)	AP (L1/L2)	APH (L1/L2)
PPBA [8]	Labeled only	✓	58.5/50.3	45.7/39.2	61.9/53.7	49.4/42.7	67.1/58.6	54.6/47.5
Pseudo labeling [4]	Labeled+Unlabeled		56.7/48.5	36.7/31.6	64.9/56.2	48.4/41.8	68.2/59.3	54.5/47.2
AutoPseudoAugment	Labeled+Unlabeled+Fused	✓	**60.3/52.1**	**48.3/41.7**	**66.5/57.8**	**55.1/47.7**	**69.6/60.8**	**58.9/51.4**

Table 4: **AutoPseudoAugment is more data efficient than SOTA auto data augmentation method (PPBA) and self-training method (Pseudo labeling).** AuotoPseudoAugment outperforms both PPBA and Pseudo labeling when trained on 10%, 30%, and 50% of the labeled training data. For vehicles, with 10% labeled run segments, AutoPseudoAugment achieves about 6 better L1 AP than others, and matches the quality of 30% labeled run segments for PPBA and Pseudo labeling. 3D detection Level 1 and 2 detection AP and APH of PointPillars model are evaluated on the Waymo Open Dataset *validation* set.

4.5 Each PseudoAugment is effective.

Previous sections show the benefit of PseudoAugments are additive to Pseudo labeling and PPBA. In this section, we train PointPillars models on 10% run segments with only one data augmentation to tease apart the contribution of

	No Aug	Common data augmentations			PseudoAugments (Ours)		
		RotateZ	FlipY	GTBBox	PseudoBBox	PseudoBackground	PseudoFrame
Vehicle	41.4	45.5 (+4.1)	44.4 (+3.0)	44.7 (+3.3)	46.4 (+5.0)	43.0 (+1.6)	45.6 (+4.2)
Pedestrian	49.1	52.7 (+3.6)	52.0 (+2.9)	50.4 (+1.3)	50.3 (+1.2)	52.2 (+3.1)	49.8 (+0.7)

Table 5: **Comparing PseudoAugments with common data augmentations.** PointPillars models are trained with only one data augmentation method on 10% of the labeled run segments. 3D detection Level 1 AP on Waymo Open Dataset *validation set* are reported.

each PseudoAugment. As a reference, we also show the performance of common data augmentation policies such as random global Z rotation, random global Y rotation, and ground truth bounding box data augmentations [64,8,39,30]. Compared to common data augmentation methods, standalone PseudoAugment achieves comparable improvements, shown in Table 5.

PseudoBBox introduces diverse foreground objects. Unlike using ground truth bounding boxes, PseudoBBox leverages unseen objects in unlabeled data to enrich the training data. On vehicle detection tasks, PseudoBBox significantly outperforms ground truth bounding box (GTBBox) augmentation (+1.7 AP), which highlights the importance of using unseen objects in unlabeled data.

PseudoBackground is important. Interestingly, we observe that utilizing the background point clouds in unlabeled data is important, especially for pedestrian detection. Taking advantage of the unseen backgrounds (PseudoBackground + 3.1 AP) is even more effective to improve model quality compared to using unseen object (PseudoBBox +1.6 AP) for detecting pedestrian.

5 Conclusion

Despite many prior works on data augmentation for 3D point clouds, data augmentation was mostly based on labeled data. In this paper, we propose to use unlabeled point clouds to augment training data and introduce PseudoAugments, which utilizes unlabeled point clouds to improve 3D detection. PseudoAugments mitigate intrinsic errors in pseudo labeled scenes while introducing diverse training data by fusing labeled and pseudo labeled scenes. We perform extensive studies and comparisons to show that PseudoAugments generalize to different architectures, model sizes, and datasets and demonstrate that AutoPseudoAugment framework outperforms existing state-of-the-art data augmentation method PPBA [8] and pseudo labeling [4] at various ratio of labeled and unlabeled data.

6 Acknowledgments

We would like to thank Yuning chai, Vijay Vasudevan, Jiquan Ngiam and the rest of Waymo and Google Brain teams for value feedback and infra supports.

References

1. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: Mixmatch: A holistic approach to semi-supervised learning. In: Advances in Neural Information Processing Systems. pp. 5049–5059 (2019) 5
2. Bewley, A., Sun, P., Mensink, T., Anguelov, D., Sminchisescu, C.: Range conditioned dilated convolutions for scale invariant 3d object detection (2020) 5
3. 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. corr abs/1903.11027 (2019) (1903) 9
4. Caine, B., Roelofs, R., Vasudevan, V., Ngiam, J., Chai, Y., Chen, Z., Shlens, J.: Pseudo-labeling for scalable 3d object detection. arXiv preprint arXiv:2103.02093 (2021) 2, 4, 5, 6, 9, 10, 11, 12, 13, 14
5. Chang, M.F., Lambert, J., Sangkloy, P., Singh, J., Bak, S., Hartnett, A., Wang, D., Carr, P., Lucey, S., Ramanan, D., et al.: Argoverse: 3d tracking and forecasting with rich maps. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8748–8757 (2019) 9
6. Chen, L.C., Lopes, R.G., Cheng, B., Collins, M.D., Cubuk, E.D., Zoph, B., Adam, H., Shlens, J.: Semi-supervised learning in video sequences for urban scene segmentation. arXiv preprint arXiv:2005.10266 (2020) 5
7. Chen, X., Ma, H., Wan, J., Li, B., Xia, T.: Multi-view 3d object detection network for autonomous driving. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1907–1915 (2017) 4, 5
8. Cheng, S., Leng, Z., Cubuk, E.D., Zoph, B., Bai, C., Ngiam, J., Song, Y., Caine, B., Vasudevan, V., Li, C., et al.: Improving 3d object detection through progressive population based augmentation. arXiv preprint arXiv:2004.00831 (2020) 2, 3, 4, 5, 6, 7, 8, 9, 12, 13, 14
9. Choi, J., Song, Y., Kwak, N.: Part-aware data augmentation for 3d object detection in point cloud. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3391–3397. IEEE (2021) 4, 5
10. Ciregan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for image classification. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. pp. 3642–3649. IEEE (2012) 4
11. Cubuk, E.D., Zoph, B., Mane, D., Vasudevan, V., Le, Q.V.: Autoaugment: Learning augmentation policies from data. arXiv preprint arXiv:1805.09501 (2018) 5
12. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 702–703 (2020) 5
13. DeVries, T., Taylor, G.W.: Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552 (2017) 4
14. Dwibedi, D., Misra, I., Hebert, M.: Cut, paste and learn: Surprisingly easy synthesis for instance detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1301–1310 (2017) 4
15. Eaton-Rosen, Z., Bragman, F., Ourselin, S., Cardoso, M.J.: Improving data augmentation for medical image segmentation (2018) 4
16. Fan, L., Pang, Z., Zhang, T., Wang, Y.X., Zhao, H., Wang, F., Wang, N., Zhang, Z.: Embracing single stride 3d object detector with sparse transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8458–8468 (2022) 5

17. Fan, L., Xiong, X., Wang, F., Wang, N., Zhang, Z.: Rangedet: In defense of range view for lidar-based 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2918–2927 (2021) 5
18. Fang, J., Zuo, X., Zhou, D., Jin, S., Wang, S., Zhang, L.: Lidar-aug: A general rendering-based augmentation framework for 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4710–4720 (2021) 4, 5
19. Ge, R., Ding, Z., Hu, Y., Wang, Y., Chen, S., Huang, L., Li, Y.: Afdet: Anchor free one stage 3d object detection (2020) 5
20. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition(CVPR) (2012) 9, 12
21. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The International Journal of Robotics Research **32**(11), 1231–1237 (2013) 9
22. Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.Y., Cubuk, E.D., Le, Q.V., Zoph, B.: Simple copy-paste is a strong data augmentation method for instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2918–2928 (2021) 5
23. Girshick, R., Radosavovic, I., Gkioxari, G., Dollár, P., He, K.: Detectron (2018) 4
24. Ho, D., Liang, E., Chen, X., Stoica, I., Abbeel, P.: Population based augmentation: Efficient learning of augmentation policy schedules. In: International Conference on Machine Learning. pp. 2731–2741. PMLR (2019) 5, 7, 8
25. Hu, P., Ziglar, J., Held, D., Ramanan, D.: What you see is what you get: Exploiting visibility for 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11001–11009 (2020) 2, 3
26. Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W.M., Donahue, J., Razavi, A., Vinyals, O., Green, T., Dunning, I., Simonyan, K., et al.: Population based training of neural networks. arXiv preprint arXiv:1711.09846 (2017) 6
27. Kahn, J., Lee, A., Hannun, A.: Self-training for end-to-end speech recognition. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 7084–7088. IEEE (2020) 5
28. Kim, S., Lee, S., Hwang, D., Lee, J., Hwang, S.J., Kim, H.J.: Point cloud augmentation with weighted local transformations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 548–557 (2021) 4
29. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (2012) 4
30. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: Pointpillars: Fast encoders for object detection from point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12697–12705 (2019) 2, 3, 4, 5, 7, 9, 11, 12, 14
31. Lee, D., Lee, J., Lee, J., Lee, H., Lee, M., Woo, S., Lee, S.: Regularization strategy for point cloud via rigidly mixed sample. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15900–15909 (2021) 4
32. Lee, D.H.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: Workshop on challenges in representation learning, ICML. vol. 3 (2013) 5
33. Lemley, J., Bazrafkan, S., Corcoran, P.: Smart augmentation learning an optimal data augmentation strategy. IEEE Access **5**, 5858–5869 (2017) 5

34. Li, R., Li, X., Heng, P.A., Fu, C.W.: Pointaugment: an auto-augmentation framework for point cloud classification. arXiv preprint arXiv:2002.10876 (2020) 4, 5
35. Lim, S., Kim, I., Kim, T., Kim, C., Kim, S.: Fast autoaugment. arXiv preprint arXiv:1905.00397 (2019) 5
36. McLachlan, G.J.: Iterative reclassification procedure for constructing an asymptotically optimal rule of allocation in discriminant analysis. Journal of the American Statistical Association **70**(350), 365–369 (1975) 5
37. Meyer, G.P., Laddha, A., Kee, E., Vallespi-Gonzalez, C., Wellington, C.K.: Lasernet: An efficient probabilistic 3d object detector for autonomous driving. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12677–12686 (2019) 5
38. Milletari, F., Navab, N., Ahmadi, S.A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. p. 565–571 (2016) 4
39. Ngiam, J., Caine, B., Han, W., Yang, B., Chai, Y., Sun, P., Zhou, Y., Yi, X., Alsharif, O., Nguyen, P., et al.: Starnet: Targeted computation for object detection in point clouds. arXiv preprint arXiv:1908.11069 (2019) 4, 5, 7, 9, 11, 14
40. Papandreou, G., Chen, L.C., Murphy, K.P., Yuille, A.L.: Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: Proceedings of the IEEE international conference on computer vision. pp. 1742–1750 (2015) 5
41. Park, D.S., Zhang, Y., Jia, Y., Han, W., Chiu, C.C., Li, B., Wu, Y., Le, Q.V.: Improved noisy student training for automatic speech recognition. arXiv preprint arXiv:2005.09629 (2020) 5
42. Qi, C.R., Liu, W., Wu, C., Su, H., Guibas, L.J.: Frustum pointnets for 3d object detection from rgb-d data. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 918–927 (2018) 5
43. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017) 5
44. Qi, C.R., Zhou, Y., Najibi, M., Sun, P., Vo, K., Deng, B., Anguelov, D.: Offboard 3d object detection from point cloud sequences. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6134–6144 (2021) 2, 4, 5
45. Ratner, A.J., Ehrenberg, H., Hussain, Z., Dunnmon, J., Ré, C.: Learning to compose domain-specific transformations for data augmentation. In: Advances in Neural Information Processing Systems. pp. 3239–3249 (2017) 5
46. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation (2015) 4
47. Rosenberg, C., Hebert, M., Schneiderman, H.: Semi-supervised self-training of object detection models 5
48. Roth, H.R., Lee, C.T., Shin, H.C., Seff, A., Kim, L., Yao, J., Lu, L., Summers, R.M.: Anatomy-specific classification of medical images using deep convolutional nets. arXiv preprint arXiv:1504.04003 (2015) 4
49. Sato, I., Nishimura, H., Yokoi, K.: Apac: Augmented pattern classification with neural networks. arXiv preprint arXiv:1505.03229 (2015) 4
50. Sheshappanavar, S.V., Singh, V.V., Kambhamettu, C.: Patchaugment: Local neighborhood augmentation in point cloud classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2118–2127 (2021) 4

51. 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. arXiv preprint arXiv:1912.13192 (2019) 4
52. 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 (2019) 5
53. Shi, S., Jiang, L., Deng, J., Wang, Z., Guo, C., Shi, J., Wang, X., Li, H.: Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. arXiv preprint arXiv:2102.00463 (2021) 5
54. Shi, S., Wang, X., Li, H.: Pointrcnn: 3d object proposal generation and detection from point cloud. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–779 (2019) 5
55. Simard, P.Y., Steinkraus, D., Platt, J.C., et al.: Best practices for convolutional neural networks applied to visual document analysis. In: Proceedings of International Conference on Document Analysis and Recognition (2003) 4
56. Sohn, K., Berthelot, D., Li, C.L., Zhang, Z., Carlini, N., Cubuk, E.D., Kurakin, A., Zhang, H., Raffel, C.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. arXiv preprint arXiv:2001.07685 (2020) 5
57. 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) 5
58. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2446–2454 (2020) 1, 2, 9, 12
59. Sun, P., Wang, W., Chai, Y., Elsayed, G., Bewley, A., Zhang, X., Sminchisescu, C., Anguelov, D.: Rsn: Range sparse net for efficient, accurate lidar 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5725–5734 (2021) 5
60. Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., Fergus, R.: Regularization of neural networks using dropconnect. In: International Conference on Machine Learning. pp. 1058–1066 (2013) 4
61. Wang, H., Cong, Y., Litany, O., Gao, Y., Guibas, L.J.: 3dioumatch: Leveraging iou prediction for semi-supervised 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14615–14624 (2021) 5
62. Wei, Y., Liang, X., Chen, Y., Shen, X., Cheng, M.M., Feng, J., Zhao, Y., Yan, S.: Stc: A simple to complex framework for weakly-supervised semantic segmentation. IEEE transactions on pattern analysis and machine intelligence **39**(11), 2314–2320 (2016) 5
63. Xie, Q., Luong, M.T., Hovy, E., Le, Q.V.: Self-training with noisy student improves imagenet classification. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2020), https://arxiv.org/abs/1911.04252 5, 8
64. Yan, Y., Mao, Y., Li, B.: Second: Sparsely embedded convolutional detection. Sensors **18**(10), 3337 (2018) 2, 3, 4, 5, 7, 14
65. Yang, B., Liang, M., Urtasun, R.: Hdnet: Exploiting hd maps for 3d object detection. In: Conference on Robot Learning. pp. 146–155. PMLR (2018) 2, 3
66. Yang, B., Luo, W., Urtasun, R.: Pixor: Real-time 3d object detection from point clouds. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 7652–7660 (2018) 2, 3, 5
67. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017) 4

68. Zheng, W., Tang, W., Jiang, L., Fu, C.W.: Se-ssd: Self-ensembling single-stage object detector from point cloud. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14494–14503 (2021) 4, 5
69. Zhou, Y., Sun, P., Zhang, Y., Anguelov, D., Gao, J., Ouyang, T., Guo, J., Ngiam, J., Vasudevan, V.: End-to-end multi-view fusion for 3d object detection in lidar point clouds. In: Conference on Robot Learning. pp. 923–932 (2020) 5
70. Zhou, Y., Tuzel, O.: Voxelnet: End-to-end learning for point cloud based 3d object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4490–4499 (2018) 2, 3, 4, 5
71. Zoph, B., Ghiasi, G., Lin, T.Y., Cui, Y., Liu, H., Cubuk, E.D., Le, Q.: Rethinking pre-training and self-training. Advances in Neural Information Processing Systems **33** (2020) 5