Boosting Single-Frame 3D Object Detection by Simulating Multi-Frame Point Clouds

Wu Zheng
wuzheng@cse.cuhk.edu.hk
CSE & SHIAE, CUHK
Hong Kong, China

Li Jiang
lijiang@mpi-inf.mpg.de
Max Planck Institute
Saarbrücken, Germany

FanBin Lu
1155141511@link.cuhk.edu.hk
CSE, CUHK
Hong Kong, China

Yangyang Ye
yeyangyang@zju.edu.cn
Zhejiang University
Hang Zhou, China

Chi-Wing Fu
cwfu@cse.cuhk.edu.hk
CSE & SHIAE, CUHK
Hong Kong, China

ABSTRACT
To boost a detector for single-frame 3D object detection, we present a new approach to train it to simulate features and responses following a detector trained on multi-frame point clouds. Our approach needs multi-frame point clouds only when training the single-frame detector, and once trained, it can detect objects with only single-frame point clouds as inputs during the inference. For this purpose, we design a novel Simulated Multi-Frame Single-Stage object Detector (SMF-SSD) framework: multi-view dense object fusion to densify ground-truth objects to generate a multi-frame point cloud; self-attention voxel distillation to facilitate one-to-many knowledge transfer from multi- to single-frame voxels; multi-scale BEV feature distillation to transfer knowledge in low-level spatial and high-level semantic BEV features; and adaptive response distillation to activate single-frame responses of high confidence and accurate localization. Experimental results on the Waymo test set show that our SMF-SSD consistently outperforms all state-of-the-art single-frame 3D object detectors for all object classes of difficulty levels 1 and 2 in terms of both mAP and mAPH.

CCS CONCEPTS
• Computing methodologies → Object detection; Scene understanding; Vision for robotics.

KEYWORDS
3D object detection, point cloud, single frame, multiple frames

ACM Reference Format:
Wu Zheng, Li Jiang, FanBin Lu, Yangyang Ye, and Chi-Wing Fu. 2022. Boosting Single-Frame 3D Object Detection by Simulating Multi-Frame Point Clouds. In Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3503161.3548123

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '22, October 10–14, 2022, Lisboa, Portugal © 2022 Association for Computing Machinery. ACM ISBN 978-1-4503-9203-7/22/10 $15.00 https://doi.org/10.1145/3503161.3548123

1 INTRODUCTION
To support autonomous driving, LiDAR-produced point clouds are widely adopted in state-of-the-art 3D object detectors [14, 34, 53, 56, 62], as they are robust to provide precise depth information in varying illumination and weather conditions. Also, the decreasing cost of LiDAR sensors promotes the popularity of point clouds in self-driving perception. Yet, due to the self occlusion of objects and the limited number of laser beams produced by LiDAR sensors, object surfaces are unlikely fully covered by the laser rays.

To overcome these shortcomings of 3D sensing with LiDAR, two very recent works [31, 54] attempt to exploit multi-frame point clouds, meaning that multiple point clouds acquired over a time range are taken together for 3D object detection. By this means, we may obtain more complete points to cover the object surfaces and improve the detection performance. 3D Auto Labeling [31] proposes to link the objects localized across multiple point-cloud frames to generate dense object points to further refine the single-frame predictions, whereas 3D-MAN [54] proposes to align and aggregate multi-view features extracted from multiple point-cloud frames to enhance the single-frame features. Though these techniques greatly improve the detection precision, multi-frame detectors inevitably sacrifice the real-time efficiency, since they require tremendous computation to process a large volume of point-cloud frames.

Figure 1: Overview of our Simulated Multi-Frame Single-Stage 3D object Detector (SMF-SSD) framework—we train a single-frame SSD to simulate features and responses following a multi-frame SSD to boost the single-frame 3D detection precision without sacrificing the inference efficiency.

4848
For 3D object detection, choosing between single- and multi-frame point clouds is essentially a choice between efficiency and accuracy. In this work, we attempt to exploit the gap between the two choices, by exploring the possibility of training a single-frame detector with only single-frame point clouds as inputs, yet capable of generating features and responses similar to those from a multi-frame detector. Our insight behind the idea is that when we see a 3D object from just one single view, we may still infer a more complete picture of the object based on our prior knowledge. Hence, when we train the single-frame detector, we take a multi-frame detector as its teacher to provide it with more complete 3D knowledge about the single-frame point cloud, such that it can learn to produce reliable features and responses like the teacher.

Our main contribution is a novel Simulated Multi-Frame Single-Stage object Detector (SMF-SSD) framework (see Figure 1) to effectively train a single-frame object detector to produce features and responses similar to those from a multi-frame object detector. In detail, we first design the multi-view dense object fusion to generate light multi-frame point clouds with only dense ground-truth objects, which are aggregated from multi-view object points in sequential time frames. This mechanism avoids huge computation overhead on the massive points produced from directly concatenating all point-cloud frames. Second, we formulate the self-attention voxel distillation to build an intermediate domain on single-frame voxels with self-attention, and then transfer the knowledge in a one-to-many manner from multi-frame voxels to single-frame ones. Third, we design a multi-scale region proposal network to abstract 3D voxel features into multi-scale bird’s eye view (BEV) features, and then formulate the multi-scale BEV feature distillation on them to separately transfer the knowledge of low-level spatial features and high-level semantics. Last, we formulate the adaptive response distillation to enhance the optimization of the classification responses with high confidence and the regression responses well aligned with ground-truth bounding boxes, aiming to activate the single-frame predictions of high confidence and accurate localization.

With the above techniques, we can effectively boost the precision of a single-frame detector without slowing down its inference speed; see a detection example with our SMF-SSD in Figure 2. The evaluation on the Waymo test set also shows that our SMF-SSD outperforms prior SOTA single-frame detectors on all object classes, and sets a new SOTA performance in terms of the average precision of all classes in both difficulty levels 1 and 2.

2 RELATED WORK

3D Object Detection. Among the LiDAR-based 3D object detectors, two-stage detectors [10, 34, 36, 37, 48, 53] focus on improving the region-proposal-aligned features for a better second-stage refinement. For example, PointRCNN [36] and STD [53] use PointNet [29, 30] to aggregate raw coordinates, semantic and segmentation features of points for region proposals, while Part-A2 [37], PV-RCNN [34], and Voxel R-CNN exploit voxelization or preset grid points to reduce the representation ambiguity of region proposals. Among the single-stage detectors [14, 52, 56, 61, 62], SE-SSD [62] builds a self-ensembling framework with CIA-SSD [61] to optimize the model with both hard and soft targets. CenterPoint [56] regresses a confidence heatmap for anchor-free 3D detection.

To further improve the feature quality, LiDAR-image detectors fuse RGB images into point clouds to extract robust multi-modality features. Early works like MV3D [6] and F-PointNet [28] predict 2D region proposals to fuse image and point cloud features. 3D-CVF [57] combines the camera and LiDAR features using the cross-view spatial feature fusion strategy. PointPainting [43] and PointAugmenting [44] decorate point clouds with image segmented scores and point-wise image features, respectively. S2M2-SSD [60] transfers knowledge from LiDAR-image features and responses to a LiDAR-only detector. Besides, a few works [3, 18, 25, 27] attempt to fuse radar and image data, and many monocular/stereo works [7, 21–23, 33, 38] detect objects only on RGB images.

To exploit multi-frame point clouds, 3D Auto Labeling [31] localizes objects in each frame and fuses the points of the predicted objects across multiple frames for further refinement. 3D-MAN [54] transfers multi-frame fusion from raw points to features for enhancing the single-frame features with multi-view features. Besides, many works [5, 20, 49, 58] propose various upsampling methods to attain high-density point clouds. SPG [51] generates semantic points at predicted foreground regions to recover missing object parts and merges semantic points with original points for detection. PC-RGNN [59] designs an encoder-decoder architecture to densify the first-stage region proposals for refinement.

Knowledge distillation. Knowledge distillation [16] was originally proposed to compress large models and later became widely adopted in image classification [15, 41, 55]. Recently, a few 2D detectors [4, 9, 13, 17, 32] decouple the BEV feature maps or focus on the significant instances for distillation. In 3D object detection, [46] first claims a multi- to single-frame distillation pipeline with a two-stage 3D detector but the technical details are not given. Recently, [47] proposes a set-to-set distillation approach to transfer knowledge between object candidates. Beyond these works, our SMF-SSD comprises a family of novel distillation techniques on intermediate voxels, BEV features, and responses to transfer knowledge from multi- to single-frame detectors, and sets a new state-of-the-art performance on single-frame 3D object detection.
3 SIMULATED MULTI-FRAME SSD

3.1 Overall Framework
Figure 3 shows the pipeline of our SMF-SSD framework. First, we pre-train a multi-frame SSD (top) on multi-frame point clouds with dense ground-truth objects. We then freeze it to produce features and responses for training the single-frame SSD (bottom) on single-frame point clouds. The important goal of our framework is to train the single-frame SSD to learn to generate features and responses similar to those from the pre-trained multi-frame SSD. Here, we design three levels of distillation techniques: (i) self-attention voxel distillation (Section 3.3), (ii) multi-scale BEV feature distillation (Section 3.4), and (iii) adaptive response distillation (Section 3.5). At testing (only pink arrows on the bottom), object detection can be done using the single-frame SSD and we only need a single-frame point cloud as input.

3.2 Multi-View Dense Object Fusion
To produce reliable features and responses, we enhance the input of the multi-frame SSD with multi-frame point clouds to reduce the sparsity and ambiguity of single-frame point clouds. Yet, simply concatenating multi-frame point clouds in the same sequence leads to massive points, which may cause unaffordable computation overhead on the detector. Hence, we propose to densify only ground-truth objects and avoid the dominated background, keeping the more significant foreground information for distillation. To this end, we design the multi-view dense object fusion to generate dense ground-truth objects by multi-view composition and produce random difference between the dense point sets of the same object appearing in sequential frames to promote sample diversity.

In detail, we denote \( \{ F_k \}_{k=1}^N \) as the frames of all \( N \) point clouds in a temporal sequence, where \( N \approx 200 \). Taking the \( j \)-th ground-truth object \( o_i \) in the \( j \)-th frame \( F_j \) as an example, our fusion procedure has the following four steps. First, from all the \( N \) frames in \( \{ F_k \}_{k=1}^N \), we form \( M \) local groups of frames \( \{ G_{l,t} \}_{l=1}^M \), where \( G_l = \{ F_{t'}, F_{t' + 1}, ..., F_{t' + 4} \} \); \( M = \left\lceil \frac{N}{4} \right\rceil \), and \( t' = 5t - 4 \). For each
With the dense ground-truth objects, multi-frame voxels are able to exploit the interrelation between single-frame voxels to build an intermediate domain for better simulating multi-frame voxel features. Meanwhile, the knowledge of a multi-frame voxel can be transferred to multiple interrelated single-frame voxels by distillation. So, we adopt [42] to transform each single-frame voxel $v_i^s$ to query $q_i$, key $k_i$, and value $v_i$ of $C$ channels linearly with $W^q$, $W^k$, and $W^v$, respectively:

$$q_i = W^q \cdot v_i^s, \quad k_i = W^k \cdot v_i^s, \quad v_i = W^v \cdot v_i^s, \quad r_{ij} = W^r \cdot d_{ij},$$

where we additionally calculate the relative position $d_{ij}$ between $v_i^s$ and $v_j^s$, and encode it as a value $r_{ij}$ with $W^r$ to represent the spatial relation and geometric structure between $v_i^s$ and $v_j^s$. Next, we multiply each pair of query $q_i$ and key $k_j$ to calculate their similarity as the attention weight $a_{ij}$, in which we add $r_{ij}$ to consider the impact of spatial relation on the voxel-wise dependency:

$$a_{ij} = \frac{q_i \cdot k_j + r_{ij}}{\sqrt{C}} \rightarrow \hat{a}_{ij} = \frac{e^{a_{ij}}}{\sum_j e^{a_{ij}}},$$

where we scale the sum by dividing $\sqrt{C}$ and normalize $a_{ij}$ with softmax to get the final attention weight $\hat{a}_{ij}$. Then, we weight each value $v_j$ with $\hat{a}_{ij}$ and calculate the cumulative sum $\hat{v}_i^s$ to replace the original voxel features $v_i^s$:

$$\hat{v}_i^s = \sum_j \hat{a}_{ij} \cdot v_j^s.$$

where the intermediate-domain voxel features $\hat{v}_i^s$ can extract complementary features from all single-frame voxels in $V^s$ to maximize the consistency with each multi-frame voxel. Conversely, the semantic knowledge of each multi-frame voxel can be adaptively transferred to all single-frame voxels in a one-to-many manner.

To alleviate this issue, we design self-attention voxel distillation to consider the missing parts or exploiting similar parts in different objects to enhance one another, aiming to reduce the domain gap with the multi-frame voxels and further enhance the knowledge transfer from multi- to single-frame voxels in a one-to-many manner.

In detail, we build an intermediate domain on single-frame voxels with self attention by comparing them with the multi-frame voxels. Importantly, we focus on voxels in the ground-truth bounding boxes to comply with the fused dense objects and avoid the massive background voxels. Procedure-wise, we first select ground-truth voxels $(V^s = \{v_i^s\}_{i=1}^{N_i^s})$ in the last sparse convolution layer of single-frame SSD (denoted by superscript $s$) for distillation, as they have rich semantics and keep the original 3D spatial information. To match with $V^s$, we filter the last-layer voxels $(V^m = \{v_i^m\}_{i=1}^{N_i^m})$ of the same center coordinates as $V^s$ from the multi-frame SSD (denoted by superscript $m$). Note also that we can always find a $v_i^m$ with the same coordinates as $v_i^s$, as the single-frame point cloud is a subset of the multi-frame one, and the same goes for voxels.

Next, we build structural dependencies on single-frame voxels $V^s$ with self attention to attain an intermediate domain for better simulating multi-frame voxel features. Meanwhile, the knowledge of a multi-frame voxel can be transferred to multiple interrelated single-frame voxels by distillation. So, we adopt [42] to transform each single-frame voxel $v_i^s$ to query $q_i$, key $k_i$, and value $v_i$ of $C$ channels linearly with $W^q$, $W^k$, and $W^v$, respectively:

$$q_i = W^q \cdot v_i^s, \quad k_i = W^k \cdot v_i^s, \quad v_i = W^v \cdot v_i^s, \quad r_{ij} = W^r \cdot d_{ij},$$

where we additionally calculate the relative position $d_{ij}$ between $v_i^s$ and $v_j^s$, and encode it as a value $r_{ij}$ with $W^r$ to represent the spatial relation and geometric structure between $v_i^s$ and $v_j^s$. Next, we multiply each pair of query $q_i$ and key $k_j$ to calculate their similarity as the attention weight $a_{ij}$, in which we add $r_{ij}$ to consider the impact of spatial relation on the voxel-wise dependency:

$$a_{ij} = \frac{q_i \cdot k_j + r_{ij}}{\sqrt{C}} \rightarrow \hat{a}_{ij} = \frac{e^{a_{ij}}}{\sum_j e^{a_{ij}}},$$

where we scale the sum by dividing $\sqrt{C}$ and normalize $a_{ij}$ with softmax to get the final attention weight $\hat{a}_{ij}$. Then, we weight each value $v_j$ with $\hat{a}_{ij}$ and calculate the cumulative sum $\hat{v}_i^s$ to replace the original voxel features $v_i^s$:

$$\hat{v}_i^s = \sum_j \hat{a}_{ij} \cdot v_j^s.$$

where the intermediate-domain voxel features $\hat{v}_i^s$ can extract complementary features from all single-frame voxels in $V^s$ to maximize the consistency with each multi-frame voxel. Conversely, the semantic knowledge of each multi-frame voxel can be adaptively transferred to all single-frame voxels in a one-to-many manner.

Overall, we use eight attention heads as above and a feed-forward network (FFN) to encode the single-frame voxels with residual connections, in which we concatenate outputs of all attention heads and compress it to the same dimension as $V^s$ with a linear layer:

$$\tilde{V}^s = Attention^8 (V^s, R^s) + V^s \quad \text{and} \quad \tilde{V}^s = FFN(\tilde{V}^s) + \tilde{V}^s,$$

where $\tilde{V}^s = (\tilde{v}_i^s)_{i=1}^{N_i}$ and $\tilde{V}^s = (\tilde{v}_i^s)_{i=1}^{N_i}$ denote the output of the self attention and FFN, respectively; and $R^s$ denotes $(r_{ij})_{i,j=1}^{N_i}$. Last, we calculate the mean square loss $L_{oxl}$ between each pair of single-frame voxel $\tilde{v}_i^s$ and multi-frame voxel $v_i^m$ for distillation:

$$L_{oxl} = \frac{1}{N_i^s} \sum_{i=1}^{N_i^s} ||\tilde{v}_i^s - v_i^m||^2_2.$$
With dense points as input, the multi-frame deep features can express rich semantics to different degrees. Due to the cross-scale interactions and also rich semantics to recognize the object classes. However, the original RPN in [56] composed of two convolution blocks cannot well present the spatial and semantic features separately and build dependencies between single-frame objects or object parts for distillation, due to limited feature scales and feature dependencies. To address it, we design a new multi-scale region proposal network (MS-RPN) with more scales and cross-scale interactions, and accordingly formulate a multi-scale BEV feature distillation technique to separately transfer low-level spatial information and high-level semantics from multi- to single-frame BEV features.

As shown in Figure 5, our MS-RPN extracts BEV features in multiple scales to emphasize low-level spatial information or high-level semantics to different degrees. Due to the cross-scale interactions and large receptive fields, it can also build dependencies between more single-frame objects or object parts to better simulate multi-frame BEV features. In detail, we exploit three groups of Conv layers to produce scale difference and large stacked receptive fields, and further utilize the DeConv layers to enrich low-level features with high-level semantics for cross-scale interaction. Next, all enriched features are upsampled to the input feature shape, followed by a bottleneck block to reduce the channels. Last, we fuse all levels of features with spatial weights to attain the output. Please find more implementation details in Section 4.1. Note that our MS-RPN can only work by incorporating the multi-scale BEV feature distillation, so it cannot improve the single detector on its own.

With the bottleneck features extracted at different scales (see the orange-frame blocks in Figure 5), we formulate the multi-scale BEV feature distillation to learn the differentiated knowledge in low-level spatial and high-level semantic features:

$$L_{\text{BEV}} = \frac{4}{|B_i|} \sum_{l=1}^{L} \sum_{j}^{W} \sum_{i}^{H} \left( \sum_{f_{ij} \in B_i} \right) \| f_{ij} - f_{ij}^m \|^2_2$$

and $f_{ij} = \text{MLP}(f_{ij}^s)$.

where we transform the single-frame feature $f_{ij}^s$ to $f_{ij}^s$ with a multi-layer perception (MLP) and calculate the consistency loss between $f_{ij}$ and the multi-frame feature $f_{ij}^m$; and $i$ is an indicator function to filter $f_{ij}$ in the ground-truth bounding boxes from the set of level-$l$ features $B_i$, aiming to focus the distillation on the foreground features. Besides, the bottleneck structure can reduce the feature channels significantly to compress the feature semantics into fewer channels, avoiding distillation on excessive channels and focusing on more significant channels for efficient knowledge transfer.

3.5 Adaptive Response Distillation

Unlike the intermediate features, classification and regression responses of a detector directly impact the overall detection performance. Given a denser amount of points, multi-frame responses usually have much higher precision than the single-frame ones, so we propose to take them as the soft targets to supervise the single-frame SSD. In our framework, we design the adaptive response distillation to improve and activate the single-frame responses with high classification confidence and accurate regressed boxes.

To avoid massive background responses, we filter the ground-truth confidence heatmap $h^g$ with threshold $\tau$ to obtain the response positions close to the ground-truth object centers, i.e., $G = h^g > \tau$. With the estimated foreground positions $G$, we further calculate the adaptive weights to distill the classification responses:

$$L_{\text{rsp}}^c = \frac{1}{|G|} \sum_{t \in G} w_t^c \cdot L_{\text{sm}l1}(h_t^c - h_t^m)$$

and

$$w_t^c = \frac{h_t^c \cdot \sum_{j \in G} L_{\text{sm}l1}(h_j^c - h_j^m)}{\sum_{j \in G} L_{\text{sm}l1}(h_j^c - h_j^m)}$$

where we multiply the adaptive weight $w_t^c$ with the smooth-L1 loss ($L_{\text{sm}l1}$) between the single- ($h_t^c$) and multi-frame ($h_t^m$) predicted confidence. With the normalization, our adaptive weights keep the cumulative sum of $L_{\text{sm}l1}$ loss unchanged, while highlighting the high-confidence responses predicted by the single-frame SSD. Therefore, we can enhance the optimization of relatively accurate predictions to motivate the single-frame SSD to produce responses of high accuracy with higher priorities. Similarly, we formulate the adaptive distillation on the regression responses as

$$L_{\text{rsp}}^r = \frac{1}{|G|} \sum_{t \in G} w_t^r \cdot L_{\text{sm}l1}(h_t^r - b_t^m)$$

and

$$w_t^r = \frac{\text{IoU}(b_t^c, b_t^m) \cdot \sum_{j \in G} L_{\text{sm}l1}(b_j^r - b_j^m)}{\sum_{j \in G} \text{IoU}(b_j^c, b_j^m) \cdot L_{\text{sm}l1}(b_j^r - b_j^m)}$$

where $L_{\text{sm}l1}$ calculates the average loss of all dimensions between each pair of bounding boxes predicted by the single- ($b_t^c$) and multi-frame ($b_t^m$) SSDs, and the adaptive weight $w_t^r$ is formulated by the $L_{\text{sm}l1}$ and $\text{IoU}$ between $b_t^c$ and $b_t^m$. By this means, we can activate more accurate single-frame regressions by enhancing the optimization of the single-frame predicted boxes that are highly overlapped with the multi-frame ones. Combining Eqs. (7) and (8), we formulate the overall loss of our adaptive response distillation:

$$L_{\text{rsp}} = \pi_1 \cdot L_{\text{rsp}}^c + \pi_2 \cdot L_{\text{rsp}}^r$$
We train the single-frame SSD end-to-end by fixing the pre-trained network and Training. We use a linear layer without bias for the FFN in Equation (4). In Equation (6), the MLP consists of a convolution layer with a $1 \times 1$ kernel and the ReLU function. In Figure 5, the strides of the DeConv layer and the first-layer of $3 \times 3$Conv are set as 2, and the kernels of all Conv and DeConv layers are set as $3 \times 3$. We compress all levels of features to single channel and normalize the concatenated three channels in each position with softmax to attain the spatial weights $w_1, w_2$, and $w_3$, which are multiplied with the corresponding features for the final addition. To preprocess the input data, we set the detection range as $[-75.2, 75.2]$m for the $x$ and $y$ axes and $[-2, 4]$m for the $z$ axis, and discretize the point clouds by a voxel size of $[0.1, 0.1, 0.15]$m along the $x$, $y$, and $z$ axes, respectively. The SSD models employed in our SMF-SSD framework are built on our implemented CenterPoint and MS-RPN, and we use the ADAM optimizer with the cosine annealing learning rate to train the model with a batch size of two. We set $\tau$ as 0.1 in Section 3.5, $\pi_1$ and $\pi_2$ as 2.0 and 1.0 in Equation (8), and $\alpha, \beta, \lambda$, and $\mu$ as 2.0, 8.0, 1.0, and 1.0 in Equation (9).
Table 2: Ablation study on our proposed modules: “mf” means our multi-view dense object fusion; “voxel”, “bev”, and “rsp” denote the distillation on intermediate voxels, multi-scale BEV features, and detection responses, respectively. We report Level-2 mAPH of our SMF-SSD trained on 1/8 Waymo train set and evaluated on the complete val split.

| Distillation Type | Veh | Ped | Cyc | All |
|-------------------|-----|-----|-----|-----|
| mf voxel bev rsp  | 57.74 | 51.46 | 62.14 | 57.11 |
| ✓ ✓ ✓ | 58.99 | 51.63 | 63.12 | 57.88 |
| ✓ ✓ ✓ | 58.45 | 52.28 | 62.53 | 57.75 |
| ✓ ✓ ✓ | 58.78 | 52.07 | 62.08 | 57.64 |
| ✓ ✓ ✓ | 59.13 | 52.59 | 63.27 | 58.33 |
| ✓ ✓ ✓ | 59.71 | 54.25 | 63.40 | 59.12 |
| ✓ ✓ ✓ | 58.51 | 52.16 | 62.18 | 57.62 |

Table 3: Ablation study on self-attention voxel distillation with Level-2 mAPH reported: “all voxel” and “gt voxel” mean distilling all voxels and the voxels in ground-truth bounding boxes, respectively; “context” means more “gt voxel” by enlarging the box dimension by 0.8m; our “self attention” is the voxel encoder to replace the common MLP; and “Number” means average number of voxels for distillation per sample.

| Distillation Type | Number | Veh | Ped | Cyc | All |
|-------------------|--------|-----|-----|-----|-----|
| baseline         | -      | 57.74 | 51.46 | 62.14 | 57.11 |
| all voxel        | 37867  | 57.76 | 51.47 | 62.15 | 57.12 |
| gt voxel         | 576    | 58.18 | 51.53 | 62.44 | 57.38 |
| + context        | 1305   | 58.23 | 51.56 | 62.51 | 57.43 |
| + self attention | 1305   | 58.89 | 51.63 | 63.12 | 57.88 |

Table 4: Ablation study on multi-scale BEV feature distillation with Level-2 mAPH: we explore different stages of BEV feature distillation in original RPN and MS-RPN. Superscripts “1” and “2” mean 128 and 64 feature channels, respectively; “rm inter” means removing the cross-scale interactions in MS-RPN.

| Type | Parameters | Stage | Veh | Ped | Cyc | All |
|------|------------|-------|-----|-----|-----|-----|
| Raw RPN | 4.58M | baseline | 58.25 | 50.86 | 62.04 | 57.03 |
|         |           | late dist | 58.30 | 50.88 | 62.03 | 57.07 |
|         |           | middle dist | 58.03 | 51.87 | 62.31 | 57.40 |
|         |           | middle dist + 1 | 58.24 | 52.09 | 62.48 | 57.60 |
|         |           | + rm inter | 58.45 | 52.28 | 62.53 | 57.75 |
| MS-RPN | 2.35M | baseline | 57.74 | 51.46 | 62.14 | 57.11 |
|         |           | early dist | 58.14 | 51.53 | 62.21 | 57.19 |
|         |           | late dist | 58.03 | 51.87 | 62.31 | 57.40 |
|         |           | middle dist | 58.24 | 52.09 | 62.48 | 57.60 |
|         |           | middle dist + 1 | 58.45 | 52.28 | 62.53 | 57.75 |

Table 5: Ablation study on adaptive response distillation with Level-2 mAPH: we compare distillation on all samples and ground-truth (“gt”) samples of classification (“cls”) and regression (“loc”) responses with and without adaptive weights.

| Distillation Type | Veh | Ped | Cyc | All |
|-------------------|-----|-----|-----|-----|
| baseline         | 57.74 | 51.46 | 62.14 | 57.11 |
| all              | 57.75 | 51.44 | 62.15 | 57.11 |
| gt cls           | 57.86 | 51.67 | 62.15 | 57.22 |
| gt loc           | 57.78 | 51.46 | 62.13 | 57.12 |
| adaptive gt cls  | 58.46 | 51.89 | 62.12 | 57.49 |
| adaptive gt loc  | 57.95 | 51.74 | 62.11 | 57.26 |
| adaptive gt cls + loc | 58.78 | 52.07 | 62.08 | 57.64 |

0.77 points contribute the largest single-module improvement, indicating the necessity of enhancing the single-frame voxel features. Also, it can be seen that the mAPH gain on vehicles is the largest one among all classes, as the large-size vehicle often have more associated voxels and thus benefit more from voxel distillation.

Table 3 shows further ablation results on voxel selection and self attention, in which we use a two-layer MLP to replace the self attention in 2nd-4th rows. Comparing the first and second rows, we can see that distilling all background and foreground voxels cannot improve the mAPH effectively, as the massive background voxels hinder the foreground knowledge transfer. While comparing the second and third rows, we can see a minor average gain of 0.26 points due to distillation only on foreground voxels, in which the reduced amount of voxels also greatly alleviate the computation overhead. From the fourth row, we further see an average gain of 0.05 points by enlarging the ground-truth bounding boxes by 0.8m, thereby involving contextual voxels to train the single-frame voxels to be more context-aware. Last, self attention on foreground voxels attains an average gain of 0.45 points, showing that the intermediate domain promotes foreground knowledge transfer.

Effect of multi-scale BEV feature distillation. Comparing the first and third rows in Table 2 shows that our multi-scale BEV feature distillation improves the Level-2 mAPH by 0.71, 0.82, 0.39, and 0.64 points on vehicles, pedestrians, cyclists, and the average, respectively. Comparing the second and fifth rows in Table 2 shows that it further boosts the mAPH by 0.24, 0.96, 0.15, and 0.45 points.

4.3 Ablation Study

To conduct ablation experiments efficiently on the large-scale Waymo dataset, we follow [35, 44] to generate a small representative training set by uniformly sampling 1/8 frames from the complete training set, and evaluate all results on the complete val split. In Tables 2-6, we show the mAPH of Level 2 on all object classes to validate the effect of each module in our SMF-SSD. In Table 7, we further explore the distance impact on the performance of our SMF-SSD.

Effect of self-attention voxel distillation. As the first and second rows in Table 2 show, our self-attention voxel distillation improves the Level-2 mAPH by 1.15, 0.17, and 0.98 points on vehicles, pedestrians, and cyclists, respectively. The resulting average gain of gains, i.e., +1.17, +1.06, +1.07, and +1.05 points, even though it tends to have sparser points. We believe that it may be caused by the sample imbalance, as vehicle objects dominate the object samples and thus hinder the knowledge transfer for cyclist objects.

Since SMF-SSD only needs single-frame point clouds as input, it has high efficiency at inference compared to the multi-frame SSD, which needs to process multi-frame point clouds. Also, the fusion of multi-frame point clouds is often time-consuming and depends on complex calibration and synchronization; see Section 4.4 for the details. Overall, our SMF-SSD can greatly boost the detection precision by learning from the multi-frame SSD, while maintaining high efficiency as the single-frame SSD at inference.
Table 6: Ablation study on multi-view dense object fusion, in which we report the Level-2 mAPH of the multi-frame SSD trained on our fused multi-frame point clouds.

| Model          | Veh | Ped | Cyc | All |
|----------------|-----|-----|-----|-----|
| Single-Frame SSD | 57.74 | 51.46 | 62.14 | 57.11 |
| Multi-Frame SSD  | 80.26 | 72.70 | 73.11 | 75.36 |

Table 7: Distance impact on our distillation method. We report Level-2 average mAPH in different distance ranges. Our SMF-SSD achieves consistent gains in all ranges.

| Method    | [0m, 30m] | [30m, 50m] | [50m, +inf] | Full Range |
|-----------|-----------|------------|-------------|------------|
| SF-SSD    | 73.50     | 56.51      | 37.02       | 59.12      |
| SMF-SSD   | 75.32     | 54.50      | 35.05       | 57.11      |
| Gain      | +1.82     | +2.01      | +1.97       | +1.98      |

on top of the self-attention voxel distillation, showing the complementarity of the two modules. Also, the mAPH gain on pedestrians is the highest, as the reduced difference in object scales on 2D feature maps narrows the gain gap of the two distillation modules.

Table 4 shows further ablation study on the multi-scale BEV feature distillation. Comparing the first and third rows, we can see that our MS-RPN cannot improve the single detector on its own. Comparing the second and fifth rows, we can see that distilling the last-layer BEV features ("late dist") leads to an average gain of 0.33 points in our MS-RPN while almost no improvement in the original RPN, validating the effectiveness of decomposing the BEV features into multiple scales for separate distillations. We also distill multi-scales BEV features before the DeConv ("early dist") and at the middle layer of bottleneck ("middle dist"), showing that compressing the middle features into 64 channels gains more. Comparing the last two rows shows that mAPH drops when removing the cross-scale interactions in MS-RPN. Comparing the numbers in second column shows that MS-RPN has only ~ 2.5 orders of magnitude, about half of the original RPN, due to fewer feature channels.

Effect of adaptive response distillation. Comparing the first and fourth rows in Table 2, we can see that this module improves the Level-2 mAP by 1.04, 0.61, -0.06, and 0.53 points on vehicles, pedestrians, cyclists, and the average, respectively. Also, comparing the fifth and sixth rows shows that it further increases mAP by 0.58, 1.66, 0.13, and 0.79 points on top of the previous two modules, validating the importance of simulating detection responses.

Table 5 shows ablation studies on adaptive response distillation. Comparing the first two rows, we can see that distilling all response samples cannot improve the baseline precision. Also, distilling the ground-truth responses of either classification or regression gives minor improvements, as shown in the third and fourth rows. With our formulated adaptive weights, we can see decent improvement in the last three rows, showing the necessity of activating the responses of high confidence and accurate localization.

Effect of multi-view dense object fusion. We use a pre-trained single-frame SSD to replace the pre-trained multi-frame SSD as the teacher for distillation, with results shown in the last two rows in Table 2. It can be seen that our pre-trained multi-frame SSD contributes 1.2 (vehicle), 2.09 (pedestrian), 1.22 (cyclist), and 1.5 (all) more points than the pre-trained single-frame SSD. In Table 6, we can see that the mAPH of our multi-frame SSD outperforms the single-frame SSD significantly, showing that our multi-view dense object fusion is effective to produce a high-performance teacher.

Distance impact on distillation. Next, we compare the Level-2 mAP gains of our SMF-SSD in different distance ranges. From Table 7, we can see consistent mAP gains of around 2 points in all different distance ranges. Since distant multi-frame objects also have fewer points than the nearby ones after the dense fusion, the information gain of dense objects at different distances against the corresponding single-frame objects are very close.

4.4 Runtime Analysis

Table 8 presents a runtime analysis of our SMF-SSD, showing its high efficiency. As shown in the table, SMF-SSD only needs 66.23ms to detect objects on a single-frame point cloud, while the multi-frame SSD needs 96.15ms to process a multi-frame point cloud, which contains dense objects and consumes more computation. Also, fusing a multi-frame point cloud offline with multi-view dense object fusion takes around 21.34 seconds on average, so it is impractical to obtain multi-frame point clouds online for training or testing the multi-frame SSD, while our SMF-SSD offers a promising solution to exploit multi-frame data without slowing down the final inference speed. Note that the experiment was done on an Intel Xeon Silver CPU and a TITAN Xp GPU with a batch size of one.

5 CONCLUSION

We presented a novel SMF-SSD framework, showing that we can train a single-frame SSD and encourage it to produce high-quality features and responses by learning from a pre-trained multi-frame SSD. This approach requires multi-frame point clouds only during training, and once trained, our SMF-SSD can efficiently detect 3D objects at high speed with single-frame point clouds as its only input. Method-wise, we first propose a multi-view dense object fusion method to densify ground-truth objects to form a multi-frame point cloud. We then propose a family of distillation techniques to transfer knowledge in our fused multi-frame point clouds to the single-frame detector: self-attention voxel distillation from multi-to-single-frame voxels in a one-to-many manner; multi-scale BEV feature distillation for low-level spatial information and high-level semantics in BEV features; and adaptive response distillation to activate single-frame responses of high confidence and accurate localization. Experimental results on the large-scale Waymo dataset show that our SMF-SSD sets a new SOTA performance on both mAP and mAPH consistently for all object classes in both difficulty levels 1 and 2, while having high inference efficiency.

Acknowledgments. This research is supported by project MMT-p2-21 of the Shun Hing Institute of Advanced Engineering (SHIAE), The Chinese University of Hong Kong (CUHK).
REFERENCES

[1] Alex Bewley, Pei Sun, Thomas Mensink, Dragomir Anguelov, and Cristian Sminchiscu. 2021. Range Conditioned Dilated Convolutions for Scale Invariant 3D Object Detection. In CoRL.

[2] Benjamin Caine, Rebecca Roozvlijf, Vijay Vasudevan, Jiqian Ngiam, Yuning Chai, Zhifeng Chen, and Jonathan Shlens. 2021. Pseudo-labeling for scalable 3D object detection. arXiv preprint arXiv:2103.02093 (2021).

[3] Simon Chadwick, Will Maddern, and Paul Newman. 2019. Distant vehicle detection using radar and vision. In ICRA.

[4] Pengpeng Chen, Shu Liu, Hengshuang Zhao, and Jiayu Jia. 2021. Districting Knowledge via Neural View. In CVPR.

[5] Xuelin Chen, Baosuan Chen, and Niiy J. Mitra. 2020. Unpaired point cloud completion on real scans using adversarial training. In ICLR.

[6] Xiaohui Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. 2017. Multi-view 3D object detection network for autonomous driving. In CVPR.

[7] Yilun Chen, Shu Liu, Xiaoyong Shen, and Jiayu Jia. 2020. DSGNN: Deep geometry network for 3D object detection. In CVPR.

[8] Shuyang Cheng, Zhaoliang Leng, Ekin Dogus Cubuk, Barret Zoph, Chunyan Bai, Aidan G. Gomez, Lukasz Kaiser, and Ilya Sutskever. 2021. Attention is all you need. In NeurIPS.

[9] Xing Dai, Zeren Jiang, Zhao Wu, Yiping Bao, Zhicheng Wang, Si Liu, and Erjin Zhou. 2021. General Instance Distillation for Object Detection. In CVPR.

[10] Jinjiong Deng, Shaozhao Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. 2021. Voxel R-CNN: Towards High Performance Voxel-based 3D Object Detection. In AAAI.

[11] Lue Fan, Xuan Xiong, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. 2021. RangeDet: In defense of range view for LiDAR-based 3D object detection. In ICCV.

[12] Tianrui Gu, Jun Wang, Shiyi Lan, Rohan Chandra, Zuxuan Wu, Larry Davis, Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. 2017. PointNet++: Deep representation 3D object detection through progressive population based augmentation. In ECCV.

[13] Xing Dai, Zeren Jiang, Zhao Wu, Yiping Bao, Zhicheng Wang, Si Liu, and Erjin Zhou. 2021. General Instance Distillation for Object Detection. In CVPR.

[14] Jinjiong Deng, Shaozhao Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. 2021. Voxel R-CNN: Towards High Performance Voxel-based 3D Object Detection. In AAAI.

[15] Yue Wang, Alireza Fathi, Jiajun Wu, Thomas Funkhouser, and Justin Solomon. 2020. Multi-Frame to Single-Frame: Knowledge Distillation for 3D Object Detection. In ECCV.

[16] Jin Young Choi. 2019. A comprehensive overhaul of feature distillation. In ICCV.

[17] Jiqian Ngiam, Xiaolong Liu, Jin-Dong Wang, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. 2020. Scalability in perception for autonomous driving. Waymo open dataset.

[18] Jin Hyeok Yoo, Yecheol Kim, Jisong Kim, and Jun Won Choi. 2020. 3D-CVF: Multiview-3D object detection using Dynamic Graphs. In NeurIPS.

[19] Jinhyong Kim, Youngseok Kim, and Dongsuk Kum. 2020. Low-level sensor fusion for 3D object detection. In CVPR.

[20] Zijian Kang, Peizhen Zhang, Xiangyu Zhang, Jian Sun, and Nanning Zheng. 2021. Categori- cal depth distribution network for monocular 3D object detection. In CVPR.

[21] Shaozhao Shi, Chaokuo Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. 2020. PV-RCNN++: Point-Voxel Feature Set Abstraction for 3D Object Detection. In CVPR.

[22] Shaozhao Shi, Li Jiang, Jianjun Deng, Zhe Wang, Chaokuo Guo, Jianping Shi, Xiaogang Wang, and Hongsheng Li. 2021. PV-RCNN++: Point-Voxel Feature Set Abstraction With Local Vector Representation for 3D Object Detection. arXiv preprint arXiv:2102.00463 (2021).

[23] Shaozhao Shi, Xiaogang Wang, and Hongsheng Li. 2019. PointRCNN: 3D object proposal generation and detection from point cloud. In CVPR.

[24] Shaozhao Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. 2020. From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network. PAMI (2020).

[25] Xuepeng Shi, Qi Ye, Xiaozhi Chen, Chuangrong Chen, Zhiyuan Shi, and Jiquan Ngiam. 2021. 3D-MAN: 3D Manifold Learning on Point Sets for 3D Classification and Segmentation. In ICCV.

[26] Shaozhao Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. 2020. From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network. PAMI (2020).

[27] Zetong Yang, Yin Zhou, Zhifeng Chen, and Jiquan Ngiam. 2021. 3D-MAN: 3D Manifold Learning on Point Sets for 3D Classification and Segmentation. In ICCV.

[28] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 2020. 3DSSD: Point-based 3D object detection. In CVPR.

[29] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 2020. 3DSSD: Point-based 3D object detection. In ICRA.

[30] Zetong Yang, Xiaofeng Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. 2021. PointAugmenting: Cross-Modal Augmentation for 3D Object Detection. In CVPR.

[31] Ruoxiao Fu, Tian Rui, Pei Sun, Hengshuang Zhao, and Jiayu Jia. 2019. Geometry-based Distance Decomposition for Monocular 3D Object Detection. In ICCV.

[32] Sohrab Vora, Alex H. Lang, Bassam Helou, and Oscar Beijbom. 2020. Point-Painting: Sequential fusion for 3D object detection. In CVPR.

[33] Ying Zhou, Xi Yi, Ouais Alsharif, Patrick Nguyen, et al. 2019. StarNet: Targeted sequence completion for object detection in point clouds. (2019).

[34] Yue Wang, Alireza Fathi, Jiajun Wu, Thomas Funkhouser, and Justin Solomon. 2020. Multi-Frame to Single-Frame: Knowledge Distillation for 3D Object Detection. In ECCV.

[35] Shaozhao Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. 2020. From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network. PAMI (2020).

[36] Xuepeng Shi, Qi Ye, Xiaozhi Chen, Chuangrong Chen, Zhiyuan Shi, and Jiquan Ngiam. 2021. 3D-MAN: 3D Manifold Learning on Point Sets for 3D Classification and Segmentation. In ICCV.

[37] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. 2017. Multi-view 3D object detection network for autonomous driving. In CVPR.

[38] Xiaohui Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. 2017. Multi-view 3D object detection network for autonomous driving. In CVPR.