Object Re-Identification from Point Clouds

Benjamin Thérien Chengjie Huang Adrian Chow Krzysztof Czarnecki
University of Waterloo
{btherien,c.huang,adrian.hei.tung.chow,k2czarne}@uwaterloo.ca

Abstract

Object re-identification (ReID) from images plays a critical role in application domains of image retrieval (surveillance, retail analytics, etc.) and multi-object tracking (autonomous driving, robotics, etc.). However, systems that additionally or exclusively perceive the world from depth sensors are becoming more commonplace without any corresponding methods for object ReID. In this work, we fill the gap by providing the first large-scale study of object ReID from point clouds and establishing its performance relative to image ReID. To enable such a study, we create two large-scale ReID datasets with paired image and LiDAR observations and propose a lightweight matching head that can be concatenated to any set or sequence processing backbone (e.g., PointNet or ViT), creating a family of comparable object ReID networks for both modalities. Run in Siamese style, our proposed point cloud ReID networks can make thousands of pairwise comparisons in real-time (10 Hz). Our findings demonstrate that their performance increases with higher sensor resolution and approaches that of image ReID when observations are sufficiently dense. Our strongest network trained at the largest scale achieves ReID accuracy exceeding 90% for rigid objects and 85% for deformable objects (without any explicit skeleton normalization). To our knowledge, we are the first to study object re-identification from real point cloud observations. Our code is available at https://github.com/bentherien/point-cloud-reid.

1. Introduction

Re-identification from images is a core component in many application domains such as surveillance [25], retail analytics [49], autonomous driving [22, 46, 50, 51], robotics [53], and many more. Given the increasing deployment of high-resolution LiDAR sensors [2, 3, 43], especially as part of robot perception systems, the development of similar techniques for ReID from point clouds has the potential to enhance these systems with a host of new capabilities. Among them, appearance-based re-identification for multi-object tracking is, perhaps, the most impactful. For instance, in robotics, whether for navigation in complex environments or for tasks like pick-and-place, the ability to accurately identify and track multiple objects in 3D space is crucial. Moreover, multi-object tracking is essential for the safe operation of autonomous vehicles.

While strong ReID performance can be obtained from image data alone, even autonomous agents equipped with arrays of RGB sensors stand to benefit from the added redundancy, diversity, and complementarity offered by processing depth-sensor information for ReID. Despite this clear added benefit, however, the existing literature on re-identification from point cloud data is almost non-existent. We are aware of only one other work studying the problem [60], but they do so on a *synthetic* person re-identification dataset. Together with the clear motivation for leveraging ReID from point clouds for many applications, the lack of established knowledge about this problem motivates our central research question: *how effective is LiDAR-based ReID compared to camera-based ReID?*

While LiDAR sensors are devoid of the lighting challenges that affect cameras, they have unique challenges of their own. The primary difficulty is the sparsity of LiDAR scans and the lack of color and texture information compared to images. A network trained to re-identify objects from point clouds must rely solely on shape information. However, certain deformable objects can pose particular difficulty as their shape can change over time. An open question is whether reliable re-identification of pedestrians is even possible from raw LiDAR input without using explicit normalization schemes. Although such questions are simple, the lack of readily available high-quality datasets for re-identification from point clouds has been a serious impediment to research thus far. We address it in what follows by providing a simple recipe for creating point clouds re-identification datasets from large-scale autonomous driving datasets.

To the best of our knowledge, ours is the first work to investigate object re-identification from point cloud observations. Our contributions can be summarized as follows:

- We propose RTMM, a symmetric matching head for ReID from point clouds that runs in real-time and shows improved convergence and generalization com-

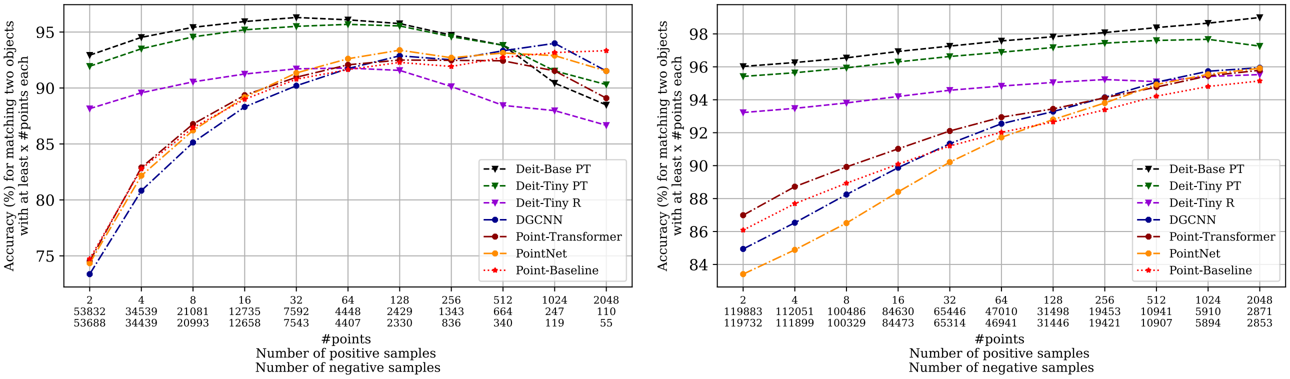

Figure 1. **The performance of point cloud ReID approaches image ReID with sufficient points.** We plot the performance of image and point cloud ReID networks as a function of point density. Left shows models trained on nuScenes and evaluated on *nuScenes Eval* set, while right shows models trained on the Waymo Open Dataset (WOD) and evaluated on *Waymo Eval*.

pared to a strong baseline.

- We provide a recipe for creating re-identification datasets from large-scale autonomous driving datasets and propose a performant training-time sampling algorithm.
- We are the first to establish point cloud ReID performance relative to image ReID on large-scale datasets. Our results demonstrate that point cloud ReID can approach the performance of image ReID when LiDAR observations are sufficiently dense.
- We fit a power-law to estimate the benefits of training beyond our compute budget, suggesting that an improvement of 2% above our strongest ReID model (89.3% ReID accuracy) is attainable with an order of magnitude more compute.

Our results outline a promising future for point-based object ReID, especially as depth-sensor resolution continues to increase.

2. Related Work

This section briefly outlines areas relevant to our study of object ReID from point clouds.

2.1. Point-Processing Networks

The ability to effectively represent irregular sets of points is essential for 3D geometry processing. Respecting the symmetries of permutation invariance (PointNet) [35] and the metric space structure of raw point clouds (PointNet++) [36] were shown early on to be important priors. Subsequent works propose edge convolutions to process point clouds in CNN-style [48], a performant and efficient residual-MLP framework [31], exploiting the benefits of depth [23], using the MLP-Mixer [4], using a transformer-based architecture [55], among others. In the following study, we select three efficient models to use for our experiments: PointNet [35], DGCNN [48], and Point-Transformer [17, 55].

2.2. 3D Single Object Tracking

Single Object Tracking (SOT) from point clouds focuses on the task of identifying a single target object within a large search area. Consequently, most methods apply point processing networks to compare the target to the search area, which can be adapted to our point cloud ReID setting. However, unlike point cloud ReID, which directly compares two objects based on shape information alone, SOT methods usually incorporate additional spatiotemporal features and motion information to aid in searching through the large search area. Many recent works [11, 16, 17, 37, 40] have shown the benefits of Siamese point-processing networks for SOT. Giancola et al. [11] learn a similarity function between cropped point cloud patches and use a Kalman filter to generate candidate bounding boxes for matching the current target observation. In follow-up work, P2B [37] eliminates the need to approximate greedy search at inference time in favor of an end-to-end regression approach that directly estimates the target's next position through Hough voting. Hui et al. [16] improve on this approach with a novel regression head inspired by work on object detection [10]. In their latest work [17], the authors improve on their previous results by employing a point-transformer backbone. Given the strong performance of their architecture for SOT, we include it in our study and show that the point transformer also attains strong performance for object re-identification.

2.3. Object Re-Identification from images and point clouds

ReID from images has been an active area of research for many years, with most work focusing on Vehicle ReID [8, 9, 12, 21, 27, 38, 39, 56, 58, 59, 63] or Person ReID [5, 13–15, 18, 19, 25, 30, 45, 54, 57, 61, 62]. In contrast, ReID from point clouds has received relatively little attention. A number of works consider ReID from RGB-D data [24, 26, 32, 34], leveraging image and depth informa-

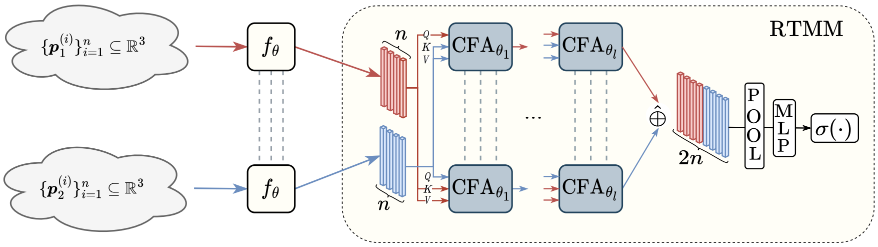
$\{\boldsymbol{p}_1^{(i)}\}_{i=1}^n \subseteq \mathbb{R}^3$
$\{\boldsymbol{p}_2^{(i)}\}_{i=1}^n \subseteq \mathbb{R}^3$
f_θ
f_θ
RTMM
n
n
Q
K
V
CFA_{θ_1}
CFA_{θ_l}
CFA_{θ_1}
CFA_{θ_l}
...
$\hat{\oplus}$
$2n$
POOL
MLP
$\sigma(\cdot)$

Figure 2. **The architecture of our proposed symmetric matching head, RTMM.** f_θ is any set or sequence processing neural network. The input is depicted as a set of n points for illustration, but could also be a sequence of image patches (e.g. for ViT). In our experiments, we use $n = 128$ points. RTMM applies l CFA blocks (linear attention) symmetrically to both inputs, allowing each to play the role of key/value and query in turn. We use $l = 2$ in our experiments. The pooling layer concatenates the max-pooled and avg-pooled representations.

tion in conjunction with skeleton normalization to improve re-identification performance in classical computer vision pipelines. In more recent work, Zheng et al. [60] propose OG-NET for pedestrian ReID from synthetic point clouds (generated from images using a human pose estimation pipeline) and RGB information. While we also use a deep neural network to process point clouds for re-identification, our study involves real observations of multiple different classes cropped directly from large-scale autonomous driving datasets.

3. Method

In the following section, we detail the architecture of our proposed Real-Time Matching Module (RTMM) for making efficient pairwise comparisons between object observations; we illustrate how existing point-based architectures can be adapted to use it, leading to a family of RTMM-based point cloud ReID networks; and we define our training objective.

3.1. A real-time matching mechanism for point sets

Given our goal of evaluating point cloud ReID in a setting that is relevant to many applications, it is important for our matching module to be capable of making a large number of pairwise comparisons (e.g., between tracks and detections from one time step to the next for multi-object tracking) in real-time. While performant architectures for comparing pairs of point clouds exist in the single object tracking literature, these methods lack an attunement to our real-time re-identification setting as they are too slow and consider an asymmetric search problem where the target and search area are not interchangeable. To construct an attuned architecture for re-identification without re-inventing the wheel, we select a state-of-the-art single object tracking method [17] and modify their "Coarse-to-Fine Correlation Network" (C2FCN), making it symmetric and real-time. We designate the resulting matching head RTMM (see fig. 2). RTMM achieves improved inference speed and generalization compared to the original C2FCN of [17](see table. 1). Moreover, RTMM's symmetric structure improves its convergence during training, allowing it to reach a lower training loss in a shorter period of time (see Figure. 3).

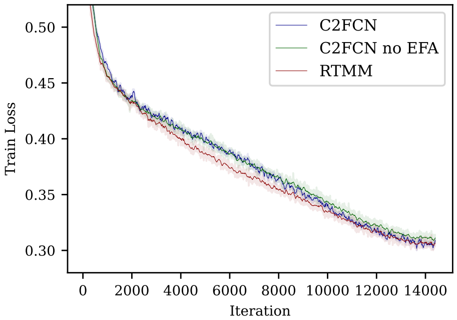

Figure 3. **The effect of different features on performance**. Each curve reports the average training loss of a two hidden-layer MLP over 10 seeds. The task is image classification on the Fashion MNIST dataset. Each optimizer was meta-trained or hyperparameter tuned on the task.

Model	Match Acc.	Inference speed	Par.
C2FCN [17]	$86.39 \pm 0.04\%$	92 ± 7.73ms	182.5k
C2FCN no EFA	$86.19 \pm 0.08\%$	$\mathbf{6.27} \pm 1.43$ms	91.3k
RTMM	$\mathbf{86.69} \pm 0.12\%$	13.2 ± 1.48ms	91.3k

Table 1. **RTMM generalizes better than other approaches while being reasonably efficient.** We train Point-Transformer models on WOD (over 4 seeds) with different match heads and evaluate their performance (matching accuracy ± standard error) on *Waymo Eval*. Inference speed is measured for a batch of 512 examples on an RTX 3090 GPU.

Starting from C2FCN, we improve the module's runtime by eliminating the ego feature augmentation module. We found that its memory and computational complexity scales poorly to a large number of comparisons (as is required for real-time applications).With the ego feature augmentation modules removed, only Cross-Feature Augmentation (CFA) modules remain, which are essentially linear attention blocks [20]. While we preserve the internal CFA block structure (see sec. F), we modify the interleaving of CFA modules to make a forward pass through RTMM symmetric with respect to each input. Specifically, to make symmetric comparisons between two sets of points $\{\boldsymbol{x}_1^{(i)}\}_i^{n_1}, \{\boldsymbol{x}_2^{(i)}\}_i^{n_2}$, with $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^3$, we apply CFA blocks symmetrically to each point cloud observation's representation allowing each point cloud to play the role of key/value and query in turn:

$$\bar{\boldsymbol{X}}_1^l = \mathrm{CFA}_{\theta_l}(\bar{\boldsymbol{X}}_1^{l-1}, \boldsymbol{X}_1, \bar{\boldsymbol{X}}_2^{l-1}, \boldsymbol{X}_2) \quad (1)$$

$$\bar{\boldsymbol{X}}_2^l = \mathrm{CFA}_{\theta_l}(\bar{\boldsymbol{X}}_2^{l-1}, \boldsymbol{X}_2, \bar{\boldsymbol{X}}_1^{l-1}, \boldsymbol{X}_1). \quad (2)$$

Where the two sets of points $\{\boldsymbol{x}_1^{(i)}\}_i^{n_1}, \{\boldsymbol{x}_2^{(i)}\}_i^{n_2}$, are designated in stacked matrix form $\boldsymbol{X}_1, \boldsymbol{X}_2 \in \mathbb{R}^{n \times 3}$ (a convention we follow henceforth) and $\bar{\boldsymbol{X}}_i^0 = f_\theta(\boldsymbol{X}_i)$ with f_θ being any point processing network. After being processed by l CFA blocks in our symmetric formulation, outputs are concatenated along the sequence dimension to which we apply an invariant pooling operation: $\mathrm{pool}(\bar{\boldsymbol{X}}_1^l \hat{\oplus} \bar{\boldsymbol{X}}_2^l)$. This differs from the original setup of [17], which only allows one point cloud to play the role of the query. Finally, an MLP is applied to the pooled representation:

$$\mathrm{RTMM}_\theta^l(\boldsymbol{X}_1, \boldsymbol{X}_2) = \mathrm{MLP}_{res}(\mathrm{pool}(\bar{\boldsymbol{X}}_1^l \hat{\oplus} \bar{\boldsymbol{X}}_2^l)) \quad (3)$$

where MLP_{res} is a residual MLP block followed by a linear projection layer, mapping each output to $\mathbb{R}$, $\mathrm{pool}(\boldsymbol{x}) := \mathrm{maxpool}(\boldsymbol{x}) \oplus \mathrm{avgpool}(\boldsymbol{x})$; $\hat{\oplus}$ designates sequence/set level concatenation; and $\oplus$ designates vector concatenation of the channel dimension. In practice, we find that setting $l = 2$ is sufficient to achieve strong matching performance. We note that on all datasets and for all point models, we subsample or resample the input point cloud to contain $n = 128$ points.

3.2. Compatibility with existing point backbones

In our empirical evaluation, we select f_θ to be PointNet [35], DGCNN [48], and Point Transformer [17]. However, almost any point processing backbone can be adapted with minimal effort to use our proposed RTMM. Due to the unstructured nature of point cloud inputs, most point-processing backbones compute an intermediate representation $f_\theta(\boldsymbol{X}) \in \mathbb{R}^{B \times N \times d}$ which is equivariant to permutations of the columns of $\boldsymbol{X}$, followed by an invariant pooling layer. Such constructions preserve the set cardinality dimension N until the pooling operation, making them amenable to processing using sequence models, such as our RTMM. Therefore, many existing point backbones can be adapted to our method by extracting their representation before invariant pooling layers.

3.3. Training objective

We train our networks for object re-identification tasks using binary cross-entropy,

$$\mathcal{L}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{y}_i \cdot \log(\hat{\boldsymbol{y}}_i) + (1 - \boldsymbol{y}_i) \log(1 - \hat{\boldsymbol{y}}_i)). \quad (4)$$

4. Large scale point cloud ReID Datasets

To train our point re-identification networks, we extract object observations from the nuScenes dataset [2] and the Waymo Open Dataset (WOD) [43]. This extraction process is non-trivial and seeks to maximize the applicability of our results to downstream applications, such as multi-object tracking, that identify objects using an object detector as a first step. Here, we briefly describe the salient details.

Sensors Each dataset contains multimodal driving data captured from one (nuScenes) or multiple (WOD) LiDAR sensors and an array of camera sensors. NuScenes employs a single 360° 32-beam LiDAR, while Waymo features one 64-beam 360° 10 Hz top-mounted sensor with four additional close-proximity LiDAR sensors on the front, back, and sides of the vehicle. This means that the WOD LiDAR scans will be many times denser than their nuScenes counterparts. The situation is reversed for cameras, however. In nuScenes, there are 6 cameras that capture a full 360° view of the scene, while the WOD only has 5 cameras with a front-facing FOV of $\sim 252°$ and a corresponding blind spot behind the vehicle. These differences allow us to examine the effect of sensor resolution on ReID performance and to explore a practically relevant setting where point-based ReID trivially complements image ReID due to the camera's blind spot.

Object Extraction To simulate the noise encountered in a real tracking-by-detection setting, we extract object observations using bounding boxes predicted by 3D object detectors. For nuScenes, we use a pre-trained BEVfusion C+L model [28], while we train our own CenterPoint model [52] for 3D object detection on WOD (see sec. A.1 for details). We post-process detections by using each model's implementation of non-maximal suppression with default settings and further eliminate noisy detections by thresholding their confidence score to be above $\tau_c = 0.1$. Using the remaining detections, we extract true and false positives by matching detected bounding boxes to ground truth bounding boxes using a permissive 3D Intersection over Union (IoU) threshold of $\tau_{IoU} = 0.01$. Hungarian matching is used here to obtain a unique assignment between ground truth and true positive bounding boxes. Duplicate true positives are discarded. To extract observations from LiDAR scans, we first crop points within an object's 3D bounding box before translating and rotating them such that the 3D bounding box becomes centered at $(0, 0, 0)^\top$ and faces a canonical orientation. Note that despite this normalization step, the observations will still contain realistic noise from the object detector's prediction; that is, the object's true orientation will not necessarily be facing the canonical orientation, nor will its true center necessarily be at $(0, 0, 0)^\top$. To extract observations from images, we project the predicted 3D bounding boxes to the image plane. Depending on its relative orien-

	Backbone	Par.	Acc.	F1 Pos.	F1 Neg.	Car	Pedestrian	Bicycle	Bus	Motorcycle	Truck	FP
nuScenes Eval	**DeiT-base**^{*I}	85.7M	92.93%	92.76%	93.1%	95.07%	89.7%	89.06%	91.88%	90.29%	92.4%	96.4%
	DeiT-tiny^{*I}	5.7M	91.94%	91.89%	91.99%	94.09%	88.23%	89.16%	90.6%	89.92%	92.34%	94.01%
	DeiT-tiny^{I}	5.7M	88.15%	88.19%	88.12%	90.34%	84.42%	85.63%	86.85%	85.81%	88.52%	89.58%
	DGCNN^{L}	0.6M	73.37%	74.25%	72.42%	77.19%	63.71%	67.1%	78.98%	66.31%	80.53%	76.73%
	Pointnet^{L}	2.8M	74.35%	74.76%	73.92%	77.97%	65%	67.38%	80.4%	67.24%	81.74%	80.1%
	Point-Transformer^{L}	0.5M	74.54%	74.72%	74.35%	78.36%	64.39%	67.24%	82.62%	68.08%	82.48%	81.04%
	Point-Baseline^{L}	0.5M	74.73%	74.74%	74.72%	78.37%	65.12%	67.81%	80.99%	67.85%	82.54%	83.69%
Waymo Eval	**DeiT-base**^{*I}	85.7M	96.02%	96%	96.04%	96.84%	94.32%	95.16%	94.89%	91.46%	95.66%	97.47%
	DeiT-tiny^{*I}	5.7M	95.42%	95.43%	95.4%	96.3%	93.69%	93.06%	92.91%	92.43%	93.04%	96.33%
	DeiT-tiny^{I}	5.7M	93.22%	93.29%	93.16%	94.14%	91.38%	92.25%	89.92%	90.63%	91.21%	93.56%
	DGCNN^{L}	0.6M	84.92%	85.03%	84.81%	86.86%	80.56%	84.12%	80.99%	74.69%	89.47%	90.3%
	Pointnet^{L}	2.8M	83.41%	83.62%	83.2%	85.51%	78.77%	82.32%	80.31%	74.4%	85.73%	88.51%
	Point-Transformer^{L}	0.5M	86.99%	87.16%	86.81%	88.84%	82.94%	83.89%	86.46%	78.92%	88.58%	91.51%
	Point-Baseline^{L}	0.5M	86.09%	86.37%	85.79%	87.93%	82.15%	82%	84.33%	78.3%	86.36%	92.16%
Waymo Eval All	**DeiT-base**^{*I}	85.7M	82.84%	83.11%	82.57%	83.03%	82.73%	79.37%	82.44%	76.65%	80.59%	83.52%
	DeiT-tiny^{*I}	5.7M	81.09%	82.06%	80%	80.91%	81.7%	78.79%	80.39%	75.88%	79.26%	77.93%
	DeiT-tiny^{I}	5.7M	77.9%	79.83%	75.55%	77.63%	78.77%	76.22%	75.16%	71.6%	75.46%	70.2%
	DGCNN^{L}	0.6M	82.34%	82.58%	82.09%	84.64%	77.4%	81.08%	80.64%	74.58%	87.01%	87%
	Pointnet^{L}	2.8M	80.94%	81.37%	80.48%	83.2%	76.15%	80.72%	76.71%	73.99%	84.47%	84.5%
	Point-Transformer^{L}	0.5M	84.14%	84.41%	83.85%	86.24%	79.71%	80.85%	84.31%	77.37%	86.91%	87.87%
	Point-Baseline^{L}	0.5M	83.4%	83.58%	83.23%	85.36%	79.31%	79.69%	83.06%	77.37%	86.08%	89.99%

*: Pre-trained & fine-tuned on ImageNet 1k, ^{I}: using RGB data, ^{L}: using LiDAR data

Table 2. **Image vs. point cloud performance for object re-identification.** While image models outperform their point-based counterparts, the large performance improvement from nuScenes to Waymo shows that increasing LiDAR sensor resolution can lead to significant performance improvement. Pedestrians benefit the most from this increase in sensor resolution, showing that at higher resolutions even the re-identification of deformable objects is possible without any explicit skeleton normalization step. These findings show a promising future for point cloud based re-identification as LiDAR sensor resolution continues to increase.

tation,the bounding box may project to a non-rectangular shape. Therefore, we always use the smallest axis-aligned bounding box enclosing the projected shape. For bounding boxes that project to multiple images, we select the projection with the largest enclosing bounding box. To maintain object identities for re-identification we use ground truth tracking labels.

Enhancing WOD Class labels The nuScenes dataset provides a large number of class labels for their tracking dataset: car, bus, pedestrian, truck, bicycle, motorcycle, and trailer. WOD, however, provides substantially fewer tracking labels with their original dataset release: vehicle, pedestrian, and bicycle. In an effort to make the results between the two datasets more comparable, we enhance the WOD labels using their point cloud segmentation labels (released in a subsequent update to the dataset). Specifically, we annotate the objects within segmentation-annotated frames using majority voting of annotated points within their bounding boxes. Then, using the tracking labels, we propagate the new class of the object to all frames. This procedure expands the labels to include truck, bus, and motorcycle (see Fig. 6 of the supplement).

Sampling at training time At training time, one epoch constitutes one pass over every unique object in the dataset. For each object O (let c denote the class of O), we flip a coin to determine whether to sample a positive or negative pair. Positive pairs are created by sampling a second observation uniformly at random from the other observations of O, while choosing to sample a negative pair leads to another coin toss. This time, we select between sampling a false positive FP of class c' (c' denotes a false positive misclassified by the object detector as belonging to class c) or a true positive by sampling an object O' of class c other than O. In either case, we must account for point density before sampling our observation. If we were to naïvely select uniformly at random among all possible observations to create a negative pair, the distributions of point densities would be wildly different between positive and negative pairs. Intuitively, this happens because positive pairs are always sampled among observations of the same object that may be more likely to have similar numbers of points. If sampling is done naïvely, models can fit the spurious correlation created between positive samples and point density during training. To avoid this pitfall, when sampling a false or true positive observation to create a negative pair, we follow $O's$ categorical distribution over the point density buckets $[2^n, 2^{n+1})$ to select the bucket from which we then sample observations. This way, the positive and negative examples follow roughly the same point density distribution during training. Table 3 shows how this simple sampling algorithm which we call ”Even Sampling” improves over naïvely sampling uniformly at random. We additionally provide pseudocode for our training-time sampling procedure in Algorithm 1.

Model	Uniform	Even	Δ	Eval Dataset
DGCNN	71.6%	73.37%	+1.77	*Waymo Eval All*
PointNet	72.92%	74.35%	+1.43	
Point-Transformer	72.83%	74.54%	+1.71	
DGCNN	82.26%	84.92%	+2.66	*nuScenes Eval*
PointNet	81.37%	83.41%	+2.04	
Point-Transformer	84.75%	86.99%	+2.24	

Table 3. **Even sampling improves performance for all models across both datasets.**

Sampling at testing time At testing time, we sample a balanced test set of large size that can accurately estimate the performance of our models at all point densities. To accomplish this, we sample at most 10 distinct positive pairs (o_1,o_2) for each object in the test set, keeping track of their point densities (d_{o_1},d_{o_2}). Then, for each positive pair (o_1,o_2), we sample a corresponding negative pair (o_1,o_2'), where o_2' has a similar point density to o_2. We define point densities as similar if they fall within the same power-two interval: $[2^n, 2^{n+1})$. Before sampling from the nuScenes test set, we filter out observations that have fewer than two points and observations without image crops. We name this test set *nuScenes Eval*. On WOD, we create two test sets. The first, called *Waymo Eval*, is created identically to *nuScenes Eval*. The second, called *Waymo Eval All*, includes all observations without any filtering. Therefore, it will include many observations that have no associated image crops as they are out of the sensor's field of view, exposing the actual performance of the image models. Table 9 of the supplement reports statistics of these evaluation sets.

5. Experiments

Our empirical evaluation is based on two re-identification datasets created from nuScenes and WOD (details provided in Sec. 4). The difference in LiDAR resolution between each dataset (32 vs. 64 beam, respectively), allows us to establish how ReID performance varies as sensor resolution increases. We also establish the relative performance of image-based and point-based ReID, show how performance varies with respect to point density within a dataset, demonstrate that increasing compute budget significantly increases our models' performance, and fit a power-law fit to extrapolate point cloud ReID performance given more compute.

5.1. Experimental Details

To place our experiments within a meaningful context, we train three image models and one point cloud baseline model to compare with our three point cloud ReID networks (PointNet [35], DGCNN [48], and Point-Transformer [17]). For our image baselines, we select DeiT [44], a family of efficient vision transformers of different sizes. They are efficient and can be adapted with little effort to use our proposed RTMM, unlike CNNs. Specifically, we choose DeiT-Tiny as our main point of comparison and train one DeiT-Tiny model from a pre-trained checkpoint and another from random initialization. DeiT-Tiny allows us to assess the performance of an image model with a *comparable* number of parameters to our point models (5M vs. 2.8M). We also train a larger DeiT-Base model from a pre-trained checkpoint for reference. To compare RTMM to another matching head as a baseline, we select C2FCN no EFA due to its real-time efficiency in combination with the point-transformer backbone.

All models were trained using identical hyperparameters and the final checkpoint is used for evaluation. We used the AdamW [29] optimizer with a learning rate of $1e{-}5$, weight decay of 0.01, cosine learning rate and momentum schedules [41], and gradient clipping of Euclidean norm 1. We use a batch size of 256×4 GPUs and 60×4 GPUs for point cloud and image experiments respectively and note that our batch normalization layers were not synchronized across devices during training. Pre-trained models are trained for 200 epochs each, while models trained from scratch are optimized for 500 epochs and 400 epochs on nuScenes and Waymo, respectively. The number of gradient descent steps for randomly initialized models is roughly the same (± 3 epochs) across both datasets as the Waymo dataset is larger.

5.2. Comparing point-based and image-based ReID

Table. 2 reports the results of our large-scale empirical study. The top section of the table corresponds to models trained on nuScenes and evaluated on *nuScenes Eval*. The bottom two sections correspond to models trained on Waymo and evaluated on *Waymo Eval* and *Waymo Eval All*, respectively. Matching accuracy is reported overall and for each individual class. We also report F1-scores for positive and negative matches. These results shed light on how point cloud ReID performance improves as sensor resolution is increased, how point-ReID performance varies for different objects and object categories (e.g. rigid vs. deformable), and the performance difference between point-based and image-based object re-identification.

When comparing the accuracy of models trained on nuScenes to those trained on WOD, we observe that there is an overall increase for all models. However, the point models improve by a much greater margin than image models: as much as 12.45% for the point transformer versus a 5.07% increase for the randomly initialized DeiT-tiny model. We hypothesize that the performance increase of image models is due to the following reasons: 1) the image sensors are of higher resolution on WOD and 2) the WOD training set is much more diverse—it has 80% more objects. This second reason is a potential confounder when assessing the extent to which the increase in point density improves point ReID performance. However, under some reasonable assumptions (see sec. C), the smaller relative increase for image models allows us to account for the confounding effect of a more diverse training set on WOD, showing that the increase in sensor resolution from nuScenes to WOD causes a performance improvement of at least $12.45\% - 5.07\% = 7.38\%$ for our point ReID models. This substantial increase in performance to 86.99% accuracy from the top-performing Point-Transformer model shows that reasonable accuracy can be obtained from point-based ReID with enough sensor resolution.

All our models on both datasets learn an unbiased matching function on aggregate as is evidenced by similar positive and negative F1 scores. When looking at class-specific results, we note that all models follow a similar increase from nuScenes to WOD as can be observed for accuracy, except

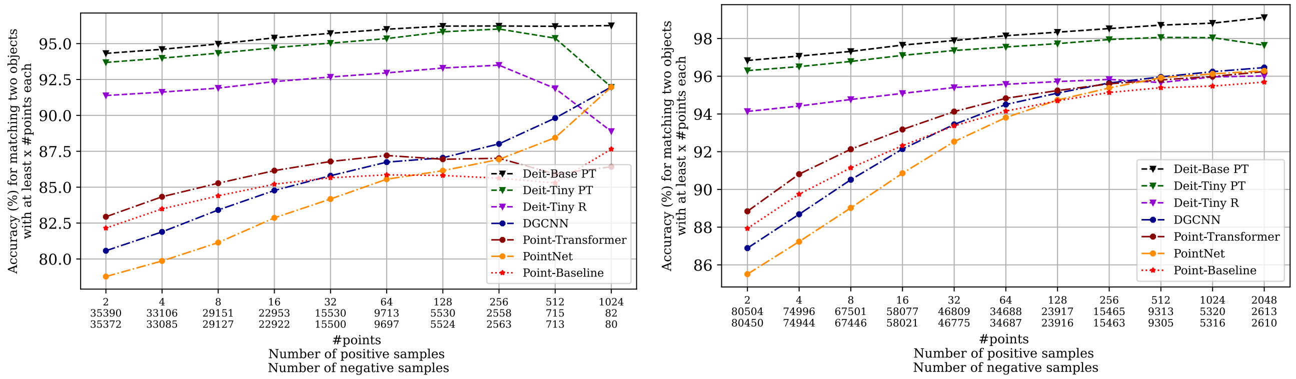

Figure 4. **Deformable vs. rigid objects.** We plot the performance of models trained on WOD and evaluated on *Waymo Eval* as a function of point density for classes pedestrian (left) and car (right). Performance on rigid objects is much stronger than for deformable objects. Our results show that ReID of deformable objects can be learned directly from data without the need of explicit skeleton normalization.

for some image models whose performance decreases on the Bus class. Of all classes, pedestrian and bicycle benefit the most from the increase in LiDAR sensor resolution with respective increases of 18.01% and 14.67%. This is a boon for point cloud ReID's applicability to downstream applications, as it shows that the re-identification of deformable objects can be learned directly from the data when sensor resolution is sufficiently large. We note that truck and bus benefit the least from increasing LiDAR sensor resolution. We hypothesize that this is because large objects will have many points regardless of the sensor's resolution.

Comparing the point re-identification models, Point-Transformer performs best on WOD, while all models perform very similarly on the nuScenes dataset. Focusing on image models exclusively, we note that the pre-trained DeiT-Base model performs best of all, as is expected given its large number of parameters. Directly comparing point models to image models, we observe that image models always outperform their point-based counterparts when observations are visible to both camera and LiDAR sensors, but that increasing sensor resolution considerably decreases this gap. When comparing the Point-Transformer to the randomly initialized DeiT-Tiny on WOD, we observe the smallest performance gap between large rigid objects (bus, truck, and car), while the smaller deformable objects (pedestrian and bicycle) pose more difficulty to the Point-Transformer and point-based models in general. This is to be expected as deformable objects create inherent shape ambiguity, which can be resolved in images by leveraging color or texture information but for point clouds, an object's shape is its primary distinguishing characteristic. While the point models perform poorer than image models overall, the relative improvement seen from nuScenes to WOD is non-trivial and suggests that the gap in performance will shrink as LiDAR sensor resolution continues to increase.

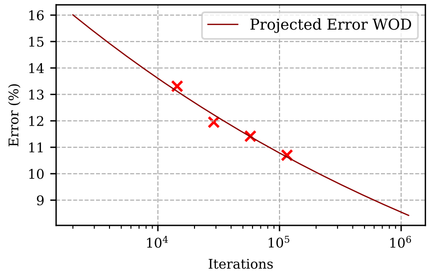

Figure 5. **Extrapolating error as a function of training iterations.** We fit a power-law, yielding $\epsilon = 34.5x^{-0.1}$.

5.3. Intra-dataset point density comparison

From our results in the previous section, it is reasonable to expect that point cloud ReID performance will continue to increase as LiDAR sensor resolution increases. Figure 1 estimates the effect that progressively higher sensor resolution has on performance by plotting the accuracy of each model as a function of point density. Specifically, the accuracy (y-axis) is measured for different subsets of *nuScenes Eval* (left) and *Waymo Eval* (right) containing pairs of point cloud observations $(\{\boldsymbol{x}_1^{(i)}\}_i^{n_1}, \{\boldsymbol{x}_2^{(i)}\}_i^{n_2})$, where $x \leq \min(n_1, n_2)$. The number of positive and negative examples for each threshold is shown on the x-axis. We observe that the magnitude of the increase is much greater for point models than image models, showing that, when sufficient points are available, point cloud ReID models can approach the performance of image re-identification models. That being said, the slope of the point ReID curves appears to decrease as higher point densities are reached, suggesting there may be a ceiling in performance. However, we do not believe that this change in slope is indicative of a ceiling in performance. Several factors such as architecture, distribution of point densities in the training set, and the number of input points can also cause a change in slope. Indeed, the difference in curve shape from nuScenes to WOD in Fig. 1 demonstrates that a denser training set increases the slope at higher point densities. Moreover, in section D.3 of the appendix, we provide further discussion of this point and empirically demonstrate that the slope improves when more input points are used (we use $n = 128$ in our experiments). Therefore, we believe that, with the appropriate

	Backbone	Epochs	Acc.	F1 Pos.	F1 Neg.	Car	Pedestrian	Bicycle	Bus	Motorcycle	Truck	FP
Waymo Eval	**Point-Transformer**^{L}	3200	89.3%	89.45%	89.16%	90.68%	86.22%	87.42%	89.21%	84.47%	90.92%	92.35%
	Point-Transformer^L	1600	88.58%	88.75%	88.41%	90.21%	85%	85.52%	87.8%	82.2%	90.18%	92.13%
	Point-Transformer^L	800	88.05%	88.2%	87.9%	89.69%	84.47%	85.09%	86.54%	81.89%	89.53%	92.14%
	Point-Transformer^L	400	86.99%	87.16%	86.81%	88.84%	82.94%	83.89%	86.46%	78.92%	88.58%	91.51%

^{L}: using LiDAR data

Table 4. **Scaling compute improves performance for all classes on WOD.**

architectural changes, the trend of increasing performance with point density will continue as LiDAR sensors that support these resolutions become available.

Figure. 4 compares the re-identification performance for deformable (pedestrian, left) and non-deformable objects (car, right) as point density is increased on *Waymo Eval*. For pedestrians, DGCNN and PointNet benefit the most from higher sensor resolutions, while the Point-Baseline and Point-Transformer models (both using a Point-Transformer backbone) are unable to take advantage of the highest point densities. This difficulty seems to be unique to deformable objects, however, as the car class follows a logarithmic trend of improvement with all models achieving similar performance. We note that the Point-Transformer model equipped with our proposed RTMM bests the Point-Baseline at every point density. With as few as 64 points per object, point-based object re-identification attains performance greater than 94% for all models when objects are rigid. This shows that point-based object re-identification can be extremely competitive in such settings.

6. Scaling training compute

As seen in Table 10 of the supplement, the number of samples in our ReID datasets is combinatorially large. For WOD, there are $4.35e8$ positive samples and $3.89e19$ negative samples. To put this in perspective it would take $\sim 13,646$ epochs to sample all possible positive samples on WOD, while we only train our models for 400 epochs. To provide practical estimates of attainable performance and showcase the best performance attainable, we train four Point-Transformer models for $400 \cdot 2^i$ epochs with $i \in \{0, 1, 2, 3\}$ on the WOD and fit a power-law through their validation accuracy.

Tables 4 and 11 show the performance of models trained on progressively larger compute budgets for WOD and nuScenes, respectively. We observe a similar effect for both datasets: performance increases across the board as the compute budget is increased. Since the largest training schedules (3200 epochs or 115200 iterations) on WOD only sample a small fraction of the enormous number of possible samples, we hypothesize that performance will continue to increase with more training.

Figure. 5 plots a power law fit to the error and training iterations from Table 4. Specifically, we fit model $\mathcal{M}_2$: $\epsilon_x - \epsilon_\infty = \beta x^c$ from [1]. The best fit obtained was $\epsilon_\infty = 0, \beta = 34.5, c = -0.1$. This suggests that even better ReID performance is attainable by continuing to increase compute with an order of magnitude more training iterations projected to yield a model with less than 9% error. This is encouraging for applications of point-based ReID which typically require low error to be worthwhile.

7. Conclusion

We have conducted the first large-scale study of object re-identifications from point cloud observations. Our findings can be summarized as follows: 1) we propose RTMM, a symmetric matching head for point cloud ReID that improves generalization and convergence; 2) we establish the performance of point cloud ReID relative to image ReID; 3) we show that our point ReID networks can attain strong ReID performance, even approaching image models, as long as the compared observations are sufficiently dense; 4) we established that point ReID performance increases as LiDAR sensor resolution is increased; and 5) we demonstrated the performance of point ReID models can be substantially increased by training for longer (89%+ accuracy).

While image ReID outperforms point ReID when observations are visible to both sensors, our results show that the latter still attains strong enough performance to be useful for downstream applications. Therefore, applications can be developed that leverage this newly discovered capability. For the time being, autonomous driving systems like the WOD vehicle, which have limited camera FOV, stand to benefit the most from the added complementarity of a ReID network processing 360° LiDAR scans. However, even vehicles equipped with cameras covering 360° can benefit from the added redundancy of point ReID, especially in cases where the observations are sufficiently dense to be reliable. In the future, as LiDAR technology continues to advance, point ReID performance can only increase—magnifying the implications of our findings. Already today, bleeding edge LiDAR sensors' feature 128 beams [42], twice the vertical resolution of WOD's top-mounted LiDAR sensor.

Our initial study opens many directions for future work. Integrating our point cloud ReID models into downstream applications such as multi-object tracking for autonomous driving or robot grasping are logical next steps. Other directions include improving ReID performance by fusing LiDAR and camera, using multi-modal fusion techniques such as [33, 47], which could work well with our framework. Finally, an important direction for future work is to design architectures that can achieve strong ReID performance at small and large point densities simultaneously. This may not be so straightforward as our results from Fig. 9 of the appendix suggest that there may be a tradeoff.

References

[1] Ibrahim M. Alabdulmohsin, Behnam Neyshabur, and Xiaohua Zhai. Revisiting neural scaling laws in language and vision. In *NeurIPS*, 2022. 8

[2] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 11618–11628. Computer Vision Foundation / IEEE, 2020. 1, 4, 14

[3] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 8748–8757. Computer Vision Foundation / IEEE, 2019. 1

[4] Jaesung Choe, Chunghyun Park, François Rameau, Jaesik Park, and In So Kweon. Pointmixer: Mlp-mixer for point cloud understanding. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXVII*, volume 13687 of *Lecture Notes in Computer Science*, pages 620–640. Springer, 2022. 2

[5] Dahjung Chung, Khalid Tahboub, and Edward J. Delp. A two stream siamese convolutional neural network for person re-identification. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 1992–2000. IEEE Computer Society, 2017. 2

[6] MMCV Contributors. MMCV: OpenMMLab computer vision foundation. https://github.com/open-mmlab/mmcv, 2018. 14

[7] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. https://github.com/open-mmlab/mmdetection3d, 2020. 14

[8] Viktor Eckstein, Arne Schumann, and Andreas Specker. Large scale vehicle re-identification by knowledge transfer from simulated data and temporal attention. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 2626–2631. Computer Vision Foundation / IEEE, 2020. 2

[9] Cunyuan Gao, Yi Hu, Yi Zhang, Rui Yao, Yong Zhou, and Jiaqi Zhao. Vehicle re-identification based on complementary features. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 2520–2526. Computer Vision Foundation / IEEE, 2020. 2

[10] Runzhou Ge, Zhuangzhuang Ding, Yihan Hu, Yu Wang, Sijia Chen, Li Huang, and Yuan Li. Afdet: Anchor free one stage 3d object detection. *CoRR*, abs/2006.12671, 2020. 2

[11] Silvio Giancola, Jesus Zarzar, and Bernard Ghanem. Leveraging shape completion for 3d siamese tracking. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 1359–1368. Computer Vision Foundation / IEEE, 2019. 2

[12] Shuting He, Hao Luo, Weihua Chen, Miao Zhang, Yuqi Zhang, Fan Wang, Hao Li, and Wei Jiang. Multi-domain learning and identity mining for vehicle re-identification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 2485–2493. Computer Vision Foundation / IEEE, 2020. 2

[13] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 14993–15002. IEEE, 2021. 2

[14] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *CoRR*, abs/1703.07737, 2017. 2

[15] Zheng Hu, Chuang Zhu, and Gang He. Hard-sample guided hybrid contrast learning for unsupervised person re-identification. In *7th IEEE International Conference on Network Intelligence and Digital Content, IC-NIDC 2021, Beijing, China, November 17-19, 2021*, pages 91–95. IEEE, 2021. 2

[16] Le Hui, Lingpeng Wang, Mingmei Cheng, Jin Xie, and Jian Yang. 3d siamese voxel-to-bev tracker for sparse point clouds. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 28714–28727, 2021. 2

[17] Le Hui, Lingpeng Wang, Linghua Tang, Kaihao Lan, Jin Xie, and Jian Yang. 3d siamese transformer network for single object tracking on point clouds. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part II*, volume 13662 of *Lecture Notes in Computer Science*, pages 293–310. Springer, 2022. 2, 3, 4, 6, 14, 20

[18] Bingliang Jiao, Lingqiao Liu, Liying Gao, Guosheng Lin, Lu Yang, Shizhou Zhang, Peng Wang, and Yanning Zhang. Dynamically transformed instance normalization network for generalizable person re-identification. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XIV*, volume 13674 of *Lecture Notes in Computer Science*, pages 285–301. Springer, 2022. 2

[19] Xin Jin, Tianyu He, Kecheng Zheng, Zhiheng Yin, Xu Shen, Zhen Huang, Ruoyu Feng, Jianqiang Huang, Zhibo Chen, and Xian-Sheng Hua. Cloth-changing person re-identification from A single image with gait prediction and regularization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 14258–14267. IEEE, 2022. 2

[20] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 5156–5165. PMLR, 2020. 3

[21] Pirazh Khorramshahi, Neehar Peri, Jun-Cheng Chen, and Rama Chellappa. The devil is in the details: Self-supervised attention for vehicle re-identification. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV*, volume 12359 of *Lecture Notes in Computer Science*, pages 369–386. Springer, 2020. 2

[22] Aleksandr Kim, Aljosa Osep, and Laura Leal-Taixé. Eagermot: 3d multi-object tracking via sensor fusion. In *IEEE International Conference on Robotics and Automation, ICRA 2021, Xi'an, China, May 30 - June 5, 2021*, pages 11315–11321. IEEE, 2021. 1

[23] Eric-Tuan Le, Iasonas Kokkinos, and Niloy J. Mitra. Going deeper with lean point networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9500–9509. Computer Vision Foundation / IEEE, 2020. 2

[24] Daniele Liciotti, Marina Paolanti, Emanuele Frontoni, Adriano Mancini, and Primo Zingaretti. Person re-identification dataset with RGB-D camera in a top-view configuration. In Kamal Nasrollahi, Cosimo Distante, Gang Hua, Andrea Cavallaro, Thomas B. Moeslund, Sebastiano Battiato, and Qiang Ji, editors, *Video Analytics. Face and Facial Expression Recognition and Audience Measurement - Third International Workshop, VAAM 2016, and Second International Workshop, FFER 2016, Cancun, Mexico, December 4, 2016, Revised Selected Papers*, volume 10165 of *Lecture Notes in Computer Science*, pages 1–11. Springer, 2016. 2

[25] Yutian Lin, Liang Zheng, Zhedong Zheng, Yu Wu, Zhilan Hu, Chenggang Yan, and Yi Yang. Improving person re-identification by attribute and identity learning. *Pattern Recognit.*, 95:151–161, 2019. 1, 2

[26] Hong Liu, Liang Hu, and Liqian Ma. Online RGB-D person re-identification based on metric model update. *CAAI Trans. Intell. Technol.*, 2(1):48–55, 2017. 2

[27] Kai Liu, Zheng Xu, Zhaohui Hou, Zhicheng Zhao, and Fei Su. Further non-local and channel attention networks for vehicle re-identification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 2494–2500. Computer Vision Foundation / IEEE, 2020. 2

[28] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. *CoRR*, abs/2205.13542, 2022. 4, 13

[29] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. 6

[30] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 1487–1495. Computer Vision Foundation / IEEE, 2019. 2

[31] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual MLP framework. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 2

[32] Matteo Munaro, Andrea Fossati, Alberto Basso, Emanuele Menegatti, and Luc Van Gool. One-shot person re-identification with a consumer depth camera. In Shaogang Gong, Marco Cristani, Shuicheng Yan, and Chen Change Loy, editors, *Person Re-Identification*, Advances in Computer Vision and Pattern Recognition, pages 161–181. Springer, 2014. 2

[33] Arsha Nagrani, Shan Yang, Anurag Arnab, Aren Jansen, Cordelia Schmid, and Chen Sun. Attention bottlenecks for multimodal fusion. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 14200–14213, 2021. 8

[34] Cosimo Patruno, Roberto Marani, Grazia Cicirelli, Ettore Stella, and Tiziana D'Orazio. People re-identification using skeleton standard posture and color descriptors from RGB-D data. *Pattern Recognit.*, 89:77–90, 2019. 2

[35] Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 77–85. IEEE Computer Society, 2017. 2, 4, 6

[36] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5099–5108, 2017. 2

[37] Haozhe Qi, Chen Feng, Zhiguo Cao, Feng Zhao, and Yang Xiao. P2B: point-to-box network for 3d object tracking in point clouds. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 6328–6337. Computer Vision Foundation / IEEE, 2020. 2

[38] Wen Qian, Hao Luo, Silong Peng, Fan Wang, Chen Chen, and Hao Li. Unstructured feature decoupling for vehicle re-identification. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European*

Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XIV, volume 13674 of *Lecture Notes in Computer Science*, pages 336–353. Springer, 2022. 2

[39] Clint Sebastian, Raffaele Imbriaco, Egor Bondarev, and Peter H. N. de With. Dual embedding expansion for vehicle re-identification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 2475–2484. Computer Vision Foundation / IEEE, 2020. 2

[40] Jiayao Shan, Sifan Zhou, Yubo Cui, and Zheng Fang. Real-time 3d single object tracking with transformer. *CoRR*, abs/2209.00860, 2022. 2

[41] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pages 369–386. SPIE, 2019. 6

[42] Autonomous Stuff. Alpha prime, powering safe autonomy. 8

[43] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 2443–2451. Computer Vision Foundation / IEEE, 2020. 1, 4

[44] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR, 2021. 6

[45] Guanshuo Wang, Yufeng Yuan, Xiong Chen, Jiwei Li, and Xi Zhou. Learning discriminative features with multiple granularities for person re-identification. In Susanne Boll, Kyoung Mu Lee, Jiebo Luo, Wenwu Zhu, Hyeran Byun, Chang Wen Chen, Rainer Lienhart, and Tao Mei, editors, *2018 ACM Multimedia Conference on Multimedia Conference, MM 2018, Seoul, Republic of Korea, October 22-26, 2018*, pages 274–282. ACM, 2018. 2

[46] Li Wang, Xinyu Zhang, Wenyuan Qin, Xiaoyu Li, Lei Yang, Zhiwei Li, Lei Zhu, Hong Wang, Jun Li, and Huaping Liu. CAMO-MOT: combined appearance-motion optimization for 3d multi-object tracking with camera-lidar fusion. *CoRR*, abs/2209.02540, 2022. 1

[47] Yikai Wang, Wenbing Huang, Fuchun Sun, Tingyang Xu, Yu Rong, and Junzhou Huang. Deep multimodal fusion by channel exchanging. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 8

[48] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph CNN for learning on point clouds. *ACM Trans. Graph.*, 38(5):146:1–146:12, 2019. 2, 4, 6

[49] Yuchen Wei, Son N. Tran, Shuxiang Xu, Byeong Ho Kang, and Matthew Springer. Deep learning for retail product recognition: Challenges and techniques. *Comput. Intell. Neurosci.*, 2020:8875910:1–8875910:23, 2020. 1

[50] Xinshuo Weng, Yongxin Wang, Yunze Man, and Kris M. Kitani. GNN3DMOT: graph neural network for 3d multi-object tracking with 2d-3d multi-feature learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 6498–6507. Computer Vision Foundation / IEEE, 2020. 1

[51] John Willes, Cody Reading, and Steven L. Waslander. Intertrack: Interaction transformer for 3d multi-object tracking. *CoRR*, abs/2208.08041, 2022. 1

[52] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 11784–11793. Computer Vision Foundation / IEEE, 2021. 4, 14

[53] Andy Zeng, Shuran Song, Kuan-Ting Yu, Elliott Donlon, Francois Robert Hogan, Maria Bauzá, Daolin Ma, Orion Taylor, Melody Liu, Eudald Romo, Nima Fazeli, Ferran Alet, Nikhil Chavan Dafle, Rachel M. Holladay, Isabella Morona, Prem Qu Nair, Druck Green, Ian H. Taylor, Weber Liu, Thomas A. Funkhouser, and Alberto Rodriguez. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. In *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*, pages 1–8. IEEE, 2018. 1

[54] Kaiwei Zeng, Munan Ning, Yaohua Wang, and Yang Guo. Hierarchical clustering with hard-batch triplet loss for person re-identification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 13654–13662. Computer Vision Foundation / IEEE, 2020. 2

[55] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H. S. Torr, and Vladlen Koltun. Point transformer. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 16239–16248. IEEE, 2021. 2

[56] Jianan Zhao, Fengliang Qi, Guangyu Ren, and Lin Xu. Phd learning: Learning with pompeiu-hausdorff distances for video-based vehicle re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 2225–2235. Computer Vision Foundation / IEEE, 2021. 2

[57] Meng Zheng, Srikrishna Karanam, Ziyan Wu, and Richard J. Radke. Re-identification with consistent attentive siamese networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June*

16-20, 2019, pages 5735–5744. Computer Vision Foundation / IEEE, 2019. 2

[58] Zhedong Zheng, Minyue Jiang, Zhigang Wang, Jian Wang, Zechen Bai, Xuanmeng Zhang, Xin Yu, Xiao Tan, Yi Yang, Shilei Wen, and Errui Ding. Going beyond real data: A robust visual representation for vehicle re-identification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 2550–2558. Computer Vision Foundation / IEEE, 2020. 2

[59] Zhedong Zheng, Tao Ruan, Yunchao Wei, Yi Yang, and Tao Mei. Vehiclenet: Learning robust visual representation for vehicle re-identification. *IEEE Trans. Multim.*, 23:2683–2693, 2021. 2

[60] Zhedong Zheng, Xiaohan Wang, Nenggan Zheng, and Yi Yang. Parameter-efficient person re-identification in the 3d space. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–14, 2022. 1, 3

[61] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2138–2147. Computer Vision Foundation / IEEE, 2019. 2

[62] Zhedong Zheng, Liang Zheng, and Yi Yang. A discriminatively learned CNN embedding for person reidentification. *ACM Trans. Multim. Comput. Commun. Appl.*, 14(1):13:1–13:20, 2018. 2

[63] Xiangyu Zhu, Zhenbo Luo, Pei Fu, and Xiang Ji. Voc-reid: Vehicle re-identification based on vehicle-orientation-camera. *CoRR*, abs/2004.09164, 2020. 2