Are NeRFs ready for autonomous driving? Towards closing the real-to-simulation gap

Carl Lindström[†,1,2] Georg Hess[†,1,2] Adam Lilja[1,2]
Maryam Fatemi[1] Lars Hammarstrand[2] Christoffer Petersson[1,2] Lennart Svensson[2]
[1]Zenseact [2]Chalmers University of Technology
{firstname.lastname}@{zenseact.com, chalmers.se}

Abstract

Neural Radiance Fields (NeRFs) have emerged as promising tools for advancing autonomous driving (AD) research, offering scalable closed-loop simulation and data augmentation capabilities. However, to trust the results achieved in simulation, one needs to ensure that AD systems perceive real and rendered data in the same way. Although the performance of rendering methods is increasing, many scenarios will remain inherently challenging to reconstruct faithfully. To this end, we propose a novel perspective for addressing the real-to-simulated data gap. Rather than solely focusing on improving rendering fidelity, we explore simple yet effective methods to enhance perception model robustness to NeRF artifacts without compromising performance on real data. Moreover, we conduct the first large-scale investigation into the real-to-simulated data gap in an AD setting using a state-of-the-art neural rendering technique. Specifically, we evaluate object detectors and an online mapping model on real and simulated data, and study the effects of different fine-tuning strategies. Our results show notable improvements in model robustness to simulated data, even improving real-world performance in some cases. Last, we delve into the correlation between the real-to-simulated gap and image reconstruction metrics, identifying FID and LPIPS as strong indicators.

1. Introduction

The development of autonomous vehicles (AVs) requires substantial and accurate testing to ensure safe behavior when deployed in the real world. In general, this has required practitioners to collect vast amounts of real-world data. Unfortunately, such collection is time-consuming and limits which safety-critical scenarios can be explored as to not risking the safety of other road users.

Neural rendering techniques, such as Neural Radiance Fields (NeRFs) [28] or Gaussian Splatting [15], provide an attractive alternative, as they can be used to simulate new scenarios in already collected data. Consequently, they enable practitioners to explore system behavior, from pixel to torque, for safety-critical scenarios that would be difficult to collect in the real world. Multiple works have recently explored how to apply NeRFs to autonomous driving (AD) data [27, 37, 44, 46, 47, 51]. With constantly increasing rendering quality, and decreasing computational demands, these methods can be expected to provide scalable and cost-effective options for offloading real-world testing.

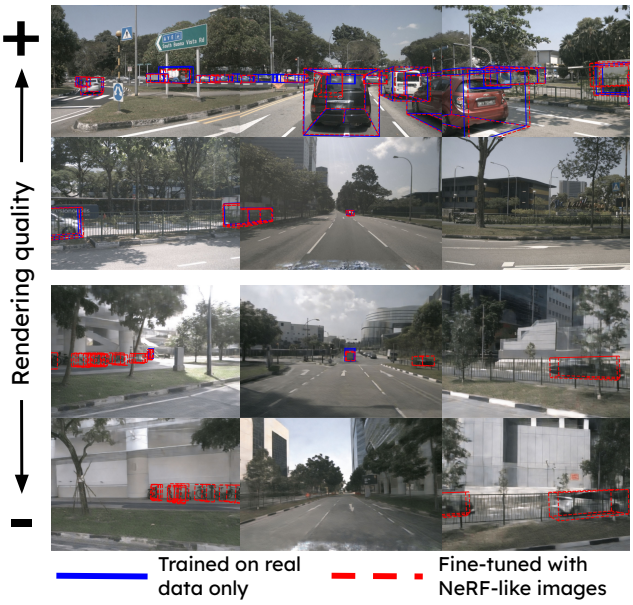


Figure 1. Using NeRFs for autonomous driving testing requires perception models to treat rendered and real images similarly. A BEVFormer model trained on real data detects objects (blue) in high-quality renderings (top). However, when quality decreases (bottom), e.g., scenes challenging for the NeRF, the same model fails to detect even close-by cars. Instead of emphasizing rendering fidelity, we propose to make models robust to these distortions. Fine-tuning the same model on NeRF-like images (red) reduces the real-to-sim gap without harming real-world performance.

[†]These authors contributed equally to this work.

Nevertheless, employing neural rendering techniques for AV testing gives rise to a fundamental concern: How reliably can conclusions drawn from simulated data be transferred to real data? To address this concern, it is essential to assess whether the system, trained on real data, interprets real and simulated data similarly, as highlighted in Fig. 1. This divergence, termed the *real2sim* gap [47], has received far less attention than its counterpart *sim2real* gap, which pertains to transferring knowledge acquired in simulation to the real world [11]. Traditionally, the real2sim problem has been addressed by improving the realism of rendered images. However, it is unknown how well common novel view synthesis (NVS) performance measures, such as PSNR, LPIPS [49], SSIM [40], or FID [10], correlate with a small real2sim gap, making it difficult to state what quality a given NeRF must reach to be useful for AV testing. Yet another aspect is that we are typically interested in the rendering quality when deviating from the original trajectory, *i.e.*, in a setting where such metrics cannot be calculated due to the lack of ground truth data to compare with.

In this paper, we propose a novel perspective on reducing the gap between real and simulated data for different perception modules of an autonomous system. Rather than improving upon the rendering quality, we aim to make the perception models more robust to NeRF artifacts without degrading performance on real data. We believe this direction to be complementary to increasing NeRF performance, and a potential key for making scalable, virtual AV testing a reality. As a first step in this direction, we show that even simple data augmentation techniques can have a large effect on model robustness against NeRF artifacts.

Further, we perform the first extensive real2sim gap study on a large-scale AD dataset and assess the performance of three object detectors alongside an online mapping model on both real data and data from a state-of-the-art (SOTA) neural rendering method. Our investigation encompasses the impact of diverse data augmentation techniques during training, as well as the fidelity of NeRF renderings during inference. We find that integrating such data during model fine-tuning notably enhances their robustness to simulated data and, in some cases, even elevates performance on real data. Lastly, we investigate the correlation between the real2sim-gap and image reconstruction metrics to provide insights into what matters for applying NeRFs as simulators for AD data. We find LPIPS and FID to be strong indicators of the real2sim-gap, and that our proposed augmentations reduce the sensitivity to poor view synthesis.

2. Related work

Novel view synthesis for autonomous driving: NeRFs have emerged as a promising approach for simulating AD data. In contrast to game engine-based methods, NeRFs remove the need for manual asset creation and are optimized to create sensor-realistic renderings by design. However, a key challenge for NeRFs is handling the scale and dynamics of automotive scenes. Neural Scene Graphs [32], Panoptic Neural Fields [18] and Panoptic NeRF [7] separate the background from moving actors by modeling each component with a separate, rigid, multi-layer perceptron (MLP). Still, these methods struggle with scaling to large scenes due to the limited expressiveness of the MLP. S-NeRF [46] addresses this by building upon Mip-NeRF 360 [2] to better handle unbounded scenes. However, its long training time makes it impractical to simulate many scenes. MARS [44] and UniSim [47] utilize the hash-grid representation from iNGP [29] and achieve efficient models, although with limitations on sensor configuration. NeuRAD [37] introduces efficient ways of modeling the important aspects of AD data, and achieves state-of-the-art performance across five AD datasets [1, 3, 8, 41, 45]. Some works in this domain [37, 43, 47] make efforts to tailor their evaluation to closed-loop AD simulation. Nonetheless, their testing sets are small in the context of AD perception, and their applicability to downstream tasks at a larger scale is unexplored. Studies with large simulated test sets have thus far exclusively been done using game engine-based simulators, see for instance [17], which compared to NeRFs require laboursome manual asset creation.

Perception for autonomous driving: Perception in 3D is a critical component of many autonomous driving solutions. Due to cameras' low cost and high availability, camera-only methods have been the subject of extensive research in recent years. For 3D object detection, FCOS3D [38] is a one-stage monocular object detector, building upon the 2D object detector FCOS [36], but regressing targets in 3D rather than 2D. PETR [25] further supports the multi-view setting and instead builds upon the decoder architecture from DETR [4], adding encoding points in the camera frustum into the image features. BEVFormer [20] also adopts a query-based architecture for multi-view input but encodes features into a bird's eye view (BEV) representation instead.

In addition to detecting objects in 3D, many autonomous vehicles also estimate the road elements of their surroundings, also known as online mapping [19, 21, 22, 26, 48]. MapTRv2 [22] is a current SOTA method that uses a vector-based representation for detected road elements such as lane dividers and road boundaries. Similarly to BEVFormer, MapTRv2 encodes image features and lifts them into a BEV representation. The objects' class and geometry are estimated through a DETR-like [4] transformer decoder.

Domain adaptation and multi-task learning: Domain adaptation (DA) is a field aiming to cope with issues arising when the training and evaluation data come from different distributions [6, 24]. Although different DA definitions exist, unsupervised DA is the most commonly studied version and assumes no access to labels in the target domain [42].

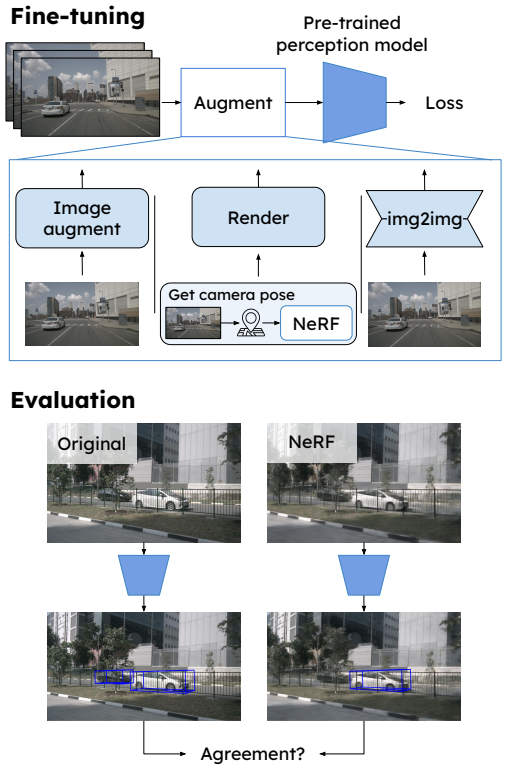


Figure 2. Overview of our data pipeline for fine-tuning (top) and evaluating (bottom) perception models. We explore three different augmentation methods for fine-tuning.

When labels exist in both the source and target domain, the problem can be categorized both as supervised domain adaptation [42] and multi-task learning [33, 50]. Multiple works study how to learn different tasks, such as semantic segmentation and depth estimation, simultaneously [13, 16, 31]. Our setting, however, differs from both these aspects, as the labels for the source and target domains are *identical*. Before neural rendering techniques, these situations rarely arose, and therefore have only been scarcely studied [47].

3. Method

Our goal is for AD systems to behave the same way when exposed to rendered and real data. As a first step in this direction, we explore how different fine-tuning strategies can make perception models more robust to artifacts in the rendered data. Specifically, given already trained models, we fine-tune the perception models using images designed to improve performance on rendered images while maintaining performance on real data, see Fig. 2. Besides reducing the real2sim gap, this can potentially lower requirements on sensor-realism, opening for a wider applicability of neural rendering methods, and lessening computational needs for training and evaluating of the said methods. Note that, while we focus on perception models, our methodology can easily be extended to end-to-end models as well [12, 14].

Last, we acknowledge that one can imagine multiple ways to achieve the goal of making models more robust, for instance by drawing inspiration from domain adaptation [6, 24] and multi-task learning [50] literature. However, fine-tuning requires minimal model-specific adjustments, allowing us to study a range of models easily.

3.1. Image augmentations

A classic strategy to obtain increased robustness to artifacts is to use image augmentations [30, 34]. Here, we select augmentations to represent various distortions present in rendered images. More specifically, we add random Gaussian noise, convolve the image with a Gaussian blur kernel, apply photometric distortions similar to the ones found in SimCLR [5], and, finally, downsample and upsample the image. The augmentations are applied sequentially, each with some probability. For reference, the perception models considered in this work are generally trained with no augmentations affecting image quality, or only photometric distortions. Details on hyperparameters can be found in the supplementary material, Appendix A.1.

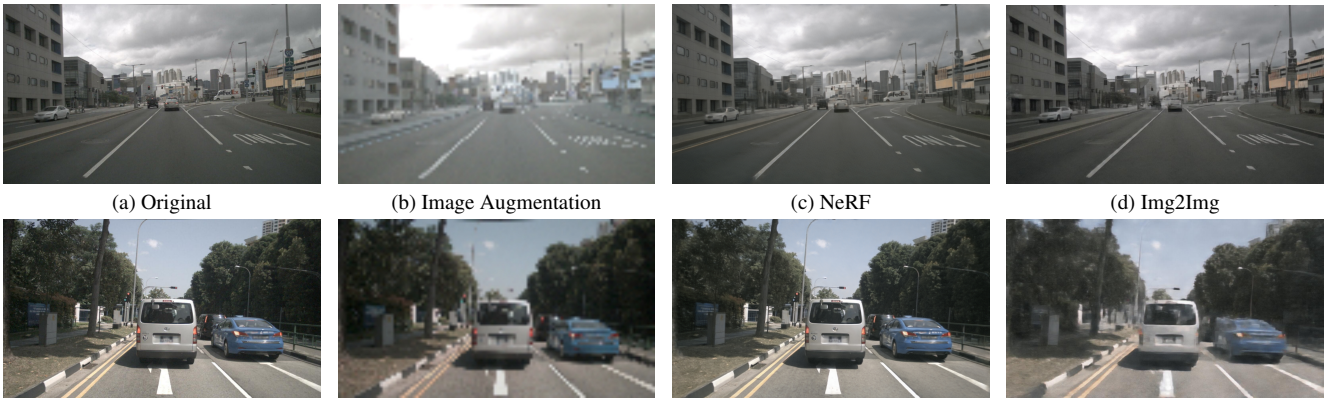
3.2. Fine-tuning with mixed-in rendered images

Another natural way to adapt perception models to NeRF-rendered data is to include such data during fine-tuning. This involves training a NeRF method on the same dataset used to supervise perception models $\mathcal{D}^{\text{real}}_{\text{train}}$. However, training NeRFs on all of $\mathcal{D}^{\text{real}}_{\text{train}}$ can be prohibitively expensive for large datasets. Instead, we train NeRFs on a subset $\mathcal{D}^{\text{nerf}} \subset \mathcal{D}^{\text{real}}_{\text{train}}$. Note that in addition to annotations for the given perception task, NeRFs for AD typically add the requirement of data in $\mathcal{D}^{\text{nerf}}$ to be sequential, where some additionally require labels for tasks such as 3D object detection [37, 47], semantic segmentation [18], or multiple types of labels [44].

Next, we divide the images for the selected sequences in $\mathcal{D}^{\text{nerf}}$ into NeRF training $\mathcal{D}^{\text{nerf}}_{\text{train}}$ and holdout $\mathcal{D}^{\text{nerf}}_{\text{fine-tune}}$ sets. Fine-tuning of the perception models is done on their entire training dataset $\mathcal{D}^{\text{real}}_{\text{train}}$, and for images that have a rendered correspondence in $\mathcal{D}^{\text{nerf}}_{\text{fine-tune}}$, we use the rendered images with probability p. This implies that the images utilized for fine-tuning have not been seen by the NeRF model.

3.3. Image-to-image translation

As mentioned previously, rendering NeRF data is an expensive data augmentation technique. Furthermore, it requires sequential data and potentially additional labeling beyond what is needed for the perception task. That is, to obtain a scalable method, we would ideally like an efficient strategy to obtain NeRF data for single images. To this end, we propose to learn to generate NeRF-like images using an image-to-image method. Given a real image, the model



(a) Original (b) Image Augmentation (c) NeRF (d) Img2Img

Figure 3. Examples of our different data augmentation strategies to make perception models more robust.

translates the image to the NeRF domain, effectively introducing typical NeRF artifacts. This enables us to vastly increase the amount of NeRF-like images during fine-tuning at a limited computational cost. We train the image-to-image model [39] using rendered images $\mathcal{D}^{\text{nerf}}_{\text{fine-tune}}$ and their corresponding real images. See Fig. 3 for visual examples of the different augmentation strategies.

4. Results

In this section, we first describe our experimental setting. We then show how the real2sim gap differs for different perception models, tasks, and augmentation strategies. Next, we study how the models with the smallest gap perceive renderings with large changes in viewpoint where no corresponding real images have been collected. Finally, we show the correlation between common metrics for novel view synthesis and the real2sim gap.

4.1. Experimental setup

We assess the real2sim gap by comparing the performance of multiple perception models when applied to real data versus their NeRF-rendered counterparts. We validate our approach to improving the robustness of perception models by implementing various augmentation strategies during fine-tuning of the models, as detailed in Sec. 3. Striving for a general approach adaptable to diverse models and tasks, we employ the same augmentation techniques across all models and tasks. Given the critical need to not sacrifice real-world performance, we also assess our augmentation methods on real data. Specifically, we test the three augmentation techniques described in Secs. 3.1 to 3.3; traditional image augmentations, rendered data, and image-to-image translation. See Appendix A.2 respectively Appendix A.4 for details regarding the datasets used during fine-tuning, and Appendix A.3 for details on training the image-to-image model.

Neural rendering: We opt to employ NeuRAD [37], the current SOTA method for NVS on AD data, as our NeRF-method. NeuRAD relies on a hashgrid representation [29] and upsampling techniques for efficient learning and inference of high-resolution data. To handle dynamic scenes, it models all actors to be rigid entities, which limits expressiveness for deformable actors such as pedestrians. Nonetheless, NeuRAD is currently the most performant NVS technique across multiple AD datasets. For the perception validation sequences, we train NeuRAD on all images except those held out for evaluating the perception models. For reference, statistics for the NeuRAD trainings are shown in Tab. 8, where we observe similar performance on images used for data augmentation as the ones used for evaluating the real2sim gap.

Dataset and tasks: Our evaluations are conducted on nuScenes, a widely recognized research ground for perception models. Compared to other large AD datasets such as Waymo Open Dataset [35], Argoverse2 [41] or Zenseact Open Dataset [1], nuScenes uses lower resolution cameras and lidars, posing an interesting challenge for neural rendering methods. We evaluate all perception models on a subset of the official validation split, namely scenes collected at daytime and without heavy rain. This is because lens flares and water spray cannot currently be handled by any neural rendering method for AD data. This filtering results in 111 scenes used for evaluation, see Appendix B.1 for details.

Further, we evaluate two tasks: 3D object detection and online mapping. These tasks consider complementary aspects of the real2sim gap, as the former focuses on foreground objects, while the latter targets the static background. Both tasks and corresponding models are described in more detail below.

3D object detection: We evaluate the gap across three camera-only 3D object detection (3DOD) models, chosen to represent different aspects of prevalent model architectures. Namely, we apply FCOS3D [38], a fully-convolutional monocular detector, PETR [25], a multi-view and 3D adaptation of DETR [4], and BEVFormer [20] a multi-view detector centered around the bird's-eye-view representation. We follow the evaluation protocol established by the

nuScenes object detection task and report mean Average Precision (mAP) and nuScenes Detection Score (NDS) on both real and rendered data. Additionally, we analyze the consistency between detections made on real and rendered data. This way, we do not only consider performance in absolute terms, but also measure if the model makes the same mistakes on both types of data. To this end, we compute NDS twice, with a distance threshold of $2\mathrm{m}$, each time treating the other set of detections as ground truth. We average the results from both evaluations to get our detection agreement (DA). All models are initialized from weights pre-trained on nuScenes and fine-tuned for a fixed number of gradient steps. See Appendix B.2 for more details on model weights and hyperparameters used for fine-tuning.

Online mapping: To extend our evaluations beyond the 3D object detection task, we also evaluate MapTRv2 [22] on the task of online mapping. Following the evaluation framework outlined in [23], we compute mAP for the classes "divider", "boundary", and "crossing". We also compute detection agreement in the same fashion as for 3DOD, alternating which set of detections is used as ground truth.

For completeness, we use both the original validation split (same as for 3DOD) and the geographically disjoint split recently proposed in [23]. In short, the original split suffers from data leakage, as there is significant geographical overlap between training and validation/testing samples. As an effect, generalization performance is largely overestimated when using the original split. Note that for the geographically disjoint split, we again remove scenes at night or with rain, resulting in 154 scenes used for evaluation. See Appendix B.1 for details.

4.2. Real2sim gap on interpolated views

We begin with studying the gap on interpolated views. These viewpoints lie in between images used to supervised NeuRAD and have corresponding real images. While these views arguably are easier to render than, for instance, shifts in the ego-vehicle position, they allow a direct comparison between real and simulated data. The real2sim gaps for 3DOD and online mapping models are reported in Tab. 1, and discussed in detail below. For all metrics, the gap is expressed as the relative performance drop compared to the real-world performance of the model without augmentations.

Gap for models without fine-tuning: Despite leveraging the current SOTA in NVS for AD data, Tab. 1 shows a significant real2sim gap across all models and tasks. Notably, there is a considerable variation in the gap among different 3DOD models, with BEVFormer exhibiting the smallest gap, whereas the mAP performance of FCOS3D is more than halved. Further, we observe a greater gap for mAP than for NDS. NDS is a weighted score where half of it consists of mAP, while the other half measures errors in terms of translation, scale, orientation, velocity, and attribute for true positives only. Considering this, the gap mainly stems from spurious or missing detection, while the quality of the true positives in terms of scale, orientation, etc., is less affected. For the online mapping task, we can see a smaller gap than for the 3DOD models, which is natural since NeuRAD often renders the static parts of a scene more accurately than the dynamic parts [37].

Image augmentations: The efficacy of basic image augmentations varies among the different models. Both BEVFormer and MapTRv2 demonstrate enhancements on simulated data while maintaining or improving performance on real data. However, FCOS3D exhibits minimal to no improvement on simulated data, despite enhancing performance on real data. In contrast, PETR displays a larger gap in terms of NDS, along with a significant decline in performance on real data. Analyzing the detection agreement shows improved consistency across the board, albeit with relatively modest improvements for FCOS3D.

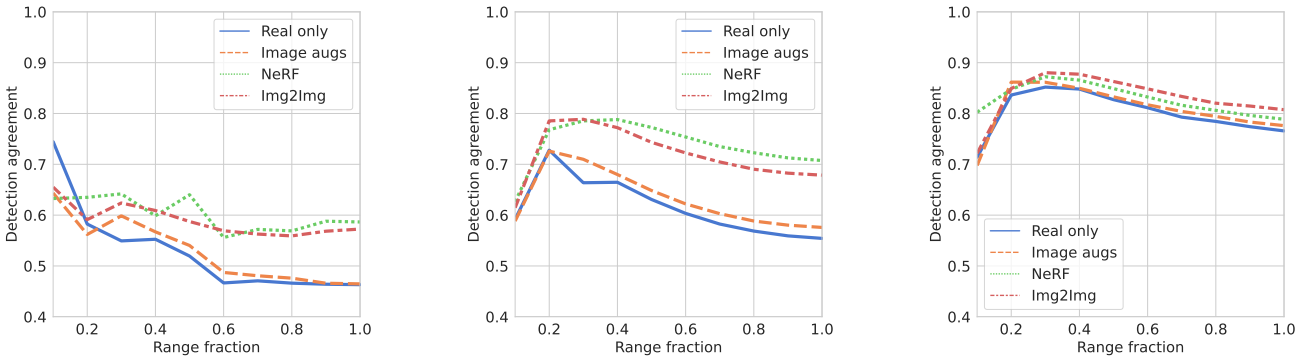
Rendered data: Incorporating rendered images during fine-tuning decreases the gap across all models. FCOS3D and PETR demonstrate substantial improvements of 74.1% respectively 45.5% in mAP on simulated data, with slight degradations on real data. Additionally, the models fine-tuned on rendered data show significant improvements in terms of detection agreement, indicating a higher consistency to the real-world detections.

Image-to-image translation: Finally, fine-tuning with image-to-image translated images leads to a significant increase in performance on simulated data across all models. It is notable that this artificial extension with NeRF-like data performs better than using the actual NeRF-data for multiple methods. However, for most methods, this also comes with some penalty in mAP performance on real data.

Detection agreement across different distances: To further gain insights into the consistency of detections relative to the detection distance, we assess the detection agreement across various fractions of the evaluation range used in the nuScenes protocol. Specifically, we examine the detection agreement for our 3D object detection models, with our different augmentations, across evaluation ranges ranging from 10% to 100% of the official protocol's evaluation range, which is 30–50m depending on the class. The results, illustrated in Fig. 4, reveal a decrease in detection agreement as evaluation distances increase, which aligns with the anticipated degradation in detection performance over longer distances. Interestingly, the fine-tunings with NeRF and image-to-image translated data notably reduce this effect for FCOS3D and PETR, as evidenced by an increasing disparity relative to the other augmentations.

Table 1. Real2sim results for the 3D object detection models and the online mapping method MapTRv2, fine-tuned with different strategies. "Sim" indicates that the model was evaluated on rendered data from NeuRAD. For online mapping, Original and Geographically Disjoint (Geogr.) refers to the splits used for training and evaluation.

Fine-tuning method	Evaluation data	3D object detection									Online mapping			
		FCOS3D			PETR			BEVFormer			Original		Geogr.	
		mAP	NDS	DA	mAP	NDS	DA	mAP	NDS	DA	mAP	DA	mAP	DA
Real data only	Real	32.2	39.8		38.6	43.1		38.4	48.5		64.5		26.6	
	Sim	13.5	28.8	46.3	20.2	31.6	55.4	29.1	42.7	76.6	54.0	71.8	23.2	67.1
	Gap (%) ↓	58.1	27.6	53.7	47.7	26.7	44.6	24.2	12.0	23.4	16.3	28.2	12.8	32.9
Image augmentations	Real	32.5	40.0		34.0	38.9		38.9	48.6		65.1		26.5	
	Sim	13.5	28.9	46.5	20.4	30.0	57.6	31.0	44.0	77.6	55.2	72.3	23.8	70.1
	Gap (%) ↓	58.1	27.4	53.5	47.2	30.4	42.4	19.3	9.3	22.4	14.4	27.7	10.5	29.9
NeRF	Real	31.2	38.6		35.1	40.0		38.5	48.3		64.9		26.9	
	Sim	23.5	33.6	58.7	29.3	37.3	70.7	31.7	44.5	78.9	56.0	74.5	24.6	70.5
	Gap (%) ↓	27.0	15.6	**41.3**	**24.1**	**13.5**	**29.3**	17.4	8.2	21.1	13.2	25.5	**7.5**	29.5
Image-to-image	Real	32.5	39.8		31.4	37.2		37.5	48.1		62.5		25.2	
	Sim	24.5	34.3	57.3	26.1	35.1	67.9	33.0	44.9	80.7	56.8	74.8	24.3	73.9
	Gap (%) ↓	**23.9**	**13.8**	42.7	32.4	18.6	32.1	**14.1**	**7.4**	**19.3**	**11.9**	**25.2**	8.3	**26.1**



(a) FCOS3D. (b) PETR. (c) BEVFormer.

Figure 4. Detection agreement vs. fraction of the evaluation range, evaluated for the 3DOD models with different fine-tuning methods.

4.3. Real2sim gap on extrapolated views

To apply a NeRF in, for instance, closed-loop simulation or data generation, it needs to produce meaningful images not only for interpolated views but even more so when extrapolating to views away from the original trajectory. To address this, we render images when laterally shifting or rotating the ego-vehicle, effectively simulating scenarios such as the vehicle performing a lane change or having departed from its original lane. We evaluate the detection agreement and perception performance by adjusting the 3D labels and detections to accommodate the specified shift or rotation. For clarity, we only evaluate models pre-trained with the image-to-image augmentation, as these displayed the best robustness in Tab. 1.

Compared to the interpolation setting, the extrapolation comes with unique challenges for evaluating the real2sim gap. First, the absence of real images in these novel scenarios prevents a comprehensive analysis of the true disparity. Instead, the field has traditionally relied on FID [9] as a performance measure on extrapolated camera views. Second, there are no assurances that the perception model would produce identical detections from the altered viewpoint, even with real data. By shifting the ego-vehicle, a scenario can become more challenging, *e.g.*, by introducing partial occlusions, or the scenario can simply be less common in the collected data, *e.g.*, images during a lane-shift. Thus, it is hard to completely disentangle these effects from the rendering quality.

Following previous work [37, 47] we render views when the ego vehicle has been moved ± 1 and ± 2 meters laterally. To ensure that the shifted views remain reasonable, *e.g.* not inside other road users or structures, we select a smaller subset of scenes and manually validate the shifts' feasibility. See Appendix B.1 for further details. The FID score and perception performance on lateral shifts can be seen in Tab. 2 and Tab. 3 for 3DOD and online mapping, re-

spectively. While the performance of all object detection methods drops as the shift increases, the ranking among the methods persists. BEVFormer is the most robust, with NDS-score dropping only 5 points from shifting the input data 2m from the original position. For the online mapping method MapTRv2, the drops are relatively large. Even shifting the ego position 1m yields a 13% and 18% drop on original and geographically disjoint splits, respectively. This discrepancy is surprising since the input data is the same as for BEVFormer and the architectures are fairly similar. By inspecting evaluation samples, we see that the model struggles with predicting rare training events, *e.g.*, the vehicle traveling slightly outside the road as Fig. 5 exemplifies.

For the rotation, the cameras are rotated around the ego-vehicle reference frame at discrete angles $\pm 5°$, $\pm 15°$, $\pm 30°$, $\pm 90°$ and $180°$. As the resulting camera positions are expected to remain within, or close to, the ego vehicle's original extent, we here use the same validation sets as in Sec. 4. Upon inspecting the renderings, they do not deteriorate noticeably with increasing rotation angle. In Tab. 4 we see similar behavior as for the lateral shift. The online mapping performance under these rotations deteriorates for each angle as we rotate further from the original pose for both the original and geographically disjoint splits.

Although the image quality degrades for larger rotations as indicated by the FID scores, we find the mapping performance to be overly sensitive to these viewpoint changes, as shown in Fig. 6. We theorize that the drop in performance also stems from these scenarios being very rare in the training data. For instance, lane changes with harsh attack angles towards a lane marker are scarce compared to in-lane driving.

To this end, we fine-tune MapTRv2 with simulated rotated views of all angles on the training data inserted into the full training set. As Tab. 4 depicts, the performance on rotated views can be improved substantially by incorporating simulated such scenarios also during training. For instance, the performance using the geographically disjoint split improves from 5.2 to 17.3 mAP on novel views perpendicular to the original poses. This is also reflected in the predictions visualized in Fig. 6. Further, it is notable that the performance on real data is improved from 26.6 to 27.5 mAP. This indicates that utilizing NeRF-rendered data also for training could be beneficial and that all performance gap is not attributed to the quality of the renderings. However, it is important to stress that disentangling how much of the performance gap stems from image quality or scenarios being outside the training distribution remains challenging.

4.4. Real2sim gap correlation to image metrics

The real2sim gap has traditionally been addressed by improving the quality of rendered images, commonly assessed

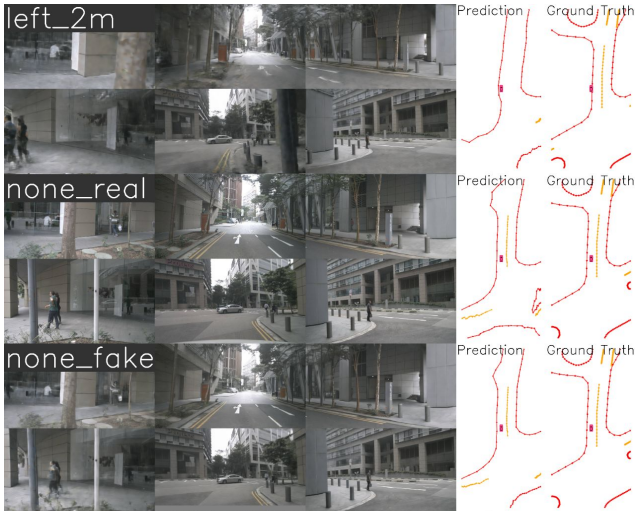


Figure 5. Online mapping predictions and ground truth for images shifted to the left (top), real images (middle), and rendered images without shift (bottom). When input data is shifted 2m to the left, the left road boundary, highlighted in red, should be straddled by the ego vehicle. However, the predictions maintain the ego vehicle within the boundary despite the image shift.

Table 2. Results for 3D object detection models fine-tuned with image-to-image translated data on laterally shifted views.

Data	Shift (m)	FID↓	FCOS3D			PETR			BEVFormer		
			mAP	NDS	DA	mAP	NDS	DA	mAP	NDS	DA
Real	0	-	30.9	38.4	100.0	35.5	40.1	100.0	35.6	45.6	100.0
Sim	0	52.0	23.3	33.1	61.6	27.5	35.0	71.3	32.3	43.7	77.4
Sim	±1	77.0	19.9	30.5	42.6	25.4	33.3	63.5	29.6	41.6	68.3
Sim	±2	95.7	16.5	28.8	39.7	22.2	31.1	53.8	26.9	39.5	59.5

Table 3. mAP results for MapTRv2 fine-tuned with image-to-image translated data evaluated on laterally shifted camera views. The performance deteriorates as the lateral shift increases for both the original and geographically disjoint (Geogr. splits).

Data	Shift (m)	FID↓	Original	Geogr.
Real	0	-	60.6	26.9
Sim	0	52.0	59.2	25.7
Sim	±1	77.0	51.5	21.2
Sim	±2	95.7	39.6	19.2

using PSNR, SSIM, LPIPS and FID under the assumption that improving these metrics reduces the gap. To test this assumption, we examine the correlation between common NVS metrics and our detection agreement. Specifically, we compute PSNR, SSIM, LPIPS and FID scores for the rendered images, and detection agreement for different augmentations applied during fine-tuning of BEVFormer. Subsequently, we aggregate the results per sequence and analyze the correlation between the aggregated data points. For the model fine-tuned with our most promising method, image-to-image, we also include FID and corresponding de-

Table 4. mAP performances for the different training methods on MapTRv2. Performance of rotated novel views can be improved substantially by injecting training data with simulated rotations.

	Finetuning	Real	Sim					
		0°	0°	±5°	±15°	±30°	±90°	180°
Original	FID↓	-	58.4	67.9	83.8	99.7	126.3	112.3
	Img2Img	62.5	56.8	53.9	35.4	19.3	6.4	19.5
	NeRF	64.9	56.0	53.1	35.3	19.3	6.8	19.1
	NeRF+Rot	64.8	56.1	57.3	51.5	43.3	35.6	36.0
Geogr.	FID↓	-	54.1	68.6	83.5	99.7	126.8	113.3
	Img2Img	25.2	24.3	22.2	17.4	11.9	5.8	12.8
	NeRF	26.9	24.6	22.6	17.3	11.4	5.2	12.1
	NeRF+Rot	27.5	25.5	24.1	21.7	18.4	17.3	16.4

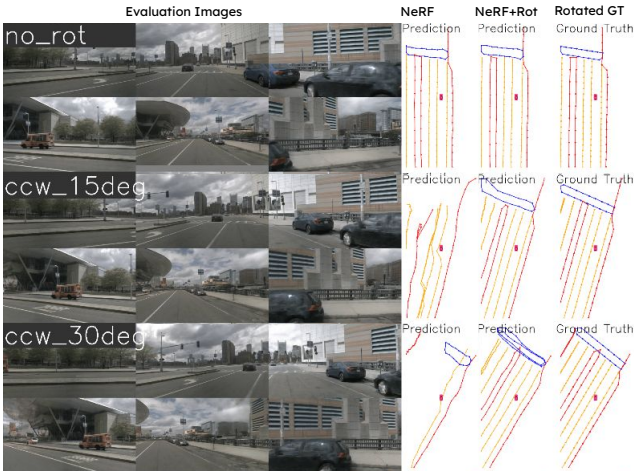
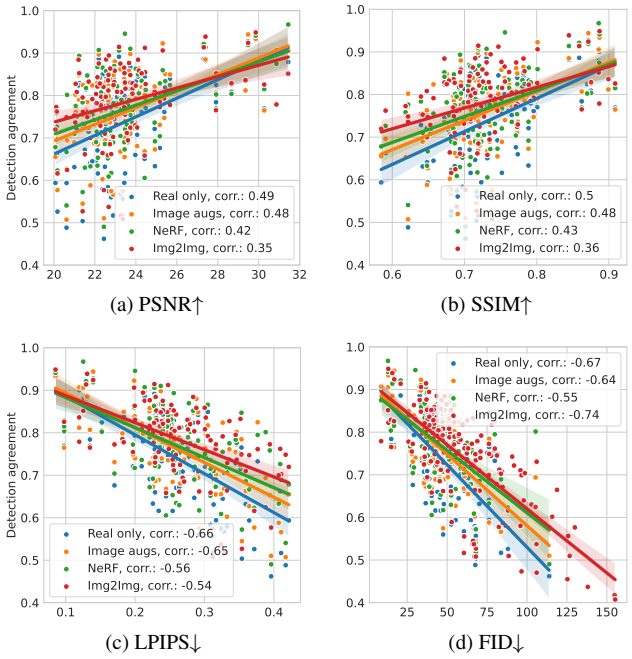


Figure 6. Online mapping predictions and ground truth for rotated input images, ccw denotes counter-clockwise. The predictions are greatly improved by injecting rotated scenarios during fine-tuning of the model.

tection agreement for the shifted scenes from Sec. 4.3, enabling us to measure the correlation for the 3DOD model and NeRF in a more practical and useful setting. Our findings, illustrated in Fig. 7 for each NVS metric and divided by augmentation, and in Fig. 9 in the supplementary material for the FID score on shifted scenes, reveal a clear correlation to detection agreement across all metrics. Notably, LPIPS and FID exhibit the strongest correlation and fewest outliers, indicating that perceptual similarity matters more to the perception model than mere reconstruction quality. Consequently, our results show that FID can be a useful indicator in the extrapolated setting where the other metrics are not applicable due to the lack of ground truth, *e.g.*, to understand how large an extrapolation can be performed for a given requirement on detection agreement. Moreover, our results indicate that, in the absence of our proposed augmentations, the model becomes considerably more sensitive to low-quality images.



(a) PSNR↑ (b) SSIM↑

(c) LPIPS↓ (d) FID↓

Figure 7. Detection agreement vs. novel view synthesis metrics for BEVFormer fine-tuned with different augmentations.

5. Conclusion

Neural rendering has emerged as a promising avenue for simulating autonomous driving (AD) data. However, to be practically useful, one must understand how the behavior of an AD system on simulated data transfers to real data. Our large-scale investigations reveal a performance gap between perception models exposed to simulated and real images. We propose a new strategy to close the gap: increasing the perception models' robustness to NeRF simulated data. We show that fine-tuning with NeRF, or NeRF-like, data substantially reduces the real2sim gap for object detection and online mapping methods with little to no performance degradation on real data. Moreover, for online mapping, we show that targeted generation of new scenarios can improve performance on real data. Nonetheless, rendering quality deteriorates rapidly when altering the ego-vehicle pose. Given our findings that low perceptual quality, *i.e.*, LPIPS and FID scores, correlate strongly with a large real2sim gap, we argue that improving rendering quality in an extrapolation setting remains a key challenge for making NeRFs useful for testing and improving AD systems.

Acknowledegments: We thank Adam Tonderski and William Ljungbergh for valuable discussions. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. Computational resources were provided by NAISS at NSC Berzelius, partially funded by the Swedish Research Council, grant agreement no. 2022-06725.

References

[1] Mina Alibeigi, William Ljungbergh, Adam Tonderski, Georg Hess, Adam Lilja, Carl Lindström, Daria Motorniuk, Junsheng Fu, Jenny Widahl, and Christoffer Petersson. Zenseact open dataset: A large-scale and diverse multimodal dataset for autonomous driving. In *ICCV*, pages 20178–20188, 2023. 2, 4

[2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, pages 5470–5479, 2022. 2

[3] 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 *CVPR*, pages 11621–11631, 2020. 2, 1

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 2, 4

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 3

[6] Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R Arabnia. A brief review of domain adaptation. *Advances in data science and information engineering: proceedings from ICDATA 2020 and IKE 2020*, pages 877–894, 2021. 2, 3

[7] Xiao Fu, Shangzhan Zhang, Tianrun Chen, Yichong Lu, Lanyun Zhu, Xiaowei Zhou, Andreas Geiger, and Yiyi Liao. Panoptic nerf: 3d-to-2d label transfer for panoptic urban scene segmentation. In *2022 International Conference on 3D Vision (3DV)*, pages 1–11. IEEE, 2022. 2

[8] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2

[9] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 6

[10] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 2

[11] Xuemin Hu, Shen Li, Tingyu Huang, Bo Tang, Rouxing Huai, and Long Chen. How simulation helps autonomous driving: A survey of sim2real, digital twins, and parallel intelligence. *IEEE Transactions on Intelligent Vehicles*, 2023. 2

[12] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17853–17862, 2023. 3

[13] Keishi Ishihara, Anssi Kanervisto, Jun Miura, and Ville Hautamaki. Multi-task learning with attention for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2902–2911, 2021. 3

[14] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350, 2023. 3

[15] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 1

[16] Apoorv Khattar, Srinidhi Hegde, and Ramya Hebbalaguppe. Cross-domain multi-task learning for object detection and saliency estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3639–3648, 2021. 3

[17] Tzofi Klinghoffer, Jonah Philion, Wenzheng Chen, Or Litany, Zan Gojcic, Jungseock Joo, Ramesh Raskar, Sanja Fidler, and Jose M Alvarez. Towards viewpoint robustness in bird's eye view segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8515–8524, 2023. 2

[18] Abhijit Kundu, Kyle Genova, Xiaoqi Yin, Alireza Fathi, Caroline Pantofaru, Leonidas J Guibas, Andrea Tagliasacchi, Frank Dellaert, and Thomas Funkhouser. Panoptic neural fields: A semantic object-aware neural scene representation. In *CVPR*, pages 12871–12881, 2022. 2, 3

[19] Qi Li, Yue Wang, Yilun Wang, and Hang Zhao. Hdmapnet: An online hd map construction and evaluation framework. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4628–4634. IEEE, 2022. 2

[20] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022. 2, 4

[21] Bencheng Liao, Shaoyu Chen, Xinggang Wang, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Chang Huang. MapTR: Structured modeling and learning for online vectorized HD map construction. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[22] Bencheng Liao, Shaoyu Chen, Yunchi Zhang, Bo Jiang, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Maptrv2: An end-to-end framework for online vectorized hd map construction. *arXiv preprint arXiv:2308.05736*, 2023. 2, 5

[23] Adam Lilja, Junsheng Fu, Erik Stenborg, and Lars Hammarstrand. Localization is all you evaluate: Data leakage in online mapping datasets and how to fix it. *arXiv preprint arXiv:2312.06420*, 2023. 5, 1, 2

[24] Xiaofeng Liu, Chaehwa Yoo, Fangxu Xing, Hyejin Oh, Georges El Fakhri, Je-Won Kang, Jonghye Woo, et al. Deep unsupervised domain adaptation: A review of recent advances and perspectives. *APSIPA Transactions on Signal and Information Processing*, 11(1), 2022. 2, 3

[25] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548. Springer, 2022. 2, 4

[26] Yicheng Liu, Tianyuan Yuan, Yue Wang, Yilun Wang, and Hang Zhao. Vectormapnet: End-to-end vectorized hd map learning. In *International Conference on Machine Learning*, pages 22352–22369. PMLR, 2023. 2

[27] William Ljungbergh, Adam Tonderski, Joakim Johnander, Holger Caesar, Kalle Åström, Michael Felsberg, and Christoffer Petersson. Neuroncap: Photorealistic closed-loop safety testing for autonomous driving. *arXiv preprint arXiv:2404.07762*, 2024. 1

[28] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 1

[29] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM TOG*, 41(4):1–15, 2022. 2, 4

[30] Kevin P Murphy. *Probabilistic machine learning: an introduction*. MIT press, 2022. 3

[31] Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. *arXiv preprint arXiv:2202.01017*, 2022. 3

[32] Julian Ost, Fahim Mannan, Nils Thuerey, Julian Knodt, and Felix Heide. Neural scene graphs for dynamic scenes. In *CVPR*, pages 2856–2865, 2021. 2

[33] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009. 3

[34] Patrice Y Simard, David Steinkraus, John C Platt, et al. Best practices for convolutional neural networks applied to visual document analysis. In *Icdar*. Edinburgh, 2003. 3

[35] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 4

[36] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: A simple and strong anchor-free object detector. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4): 1922–1933, 2020. 2

[37] Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and Christoffer Petersson. Neurad: Neural rendering for autonomous driving. *arXiv preprint arXiv:2311.15260*, 2023. 1, 2, 3, 4, 5, 6

[38] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 913–922, 2021. 2, 4

[39] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018. 4, 1

[40] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 2

[41] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*, 2021. 2, 4

[42] Garrett Wilson and Diane J Cook. A survey of unsupervised deep domain adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(5):1–46, 2020. 2, 3

[43] Hanfeng Wu, Xingxing Zuo, Stefan Leutenegger, Or Litany, Konrad Schindler, and Shengyu Huang. Dynamic lidar resimulation using compositional neural fields. *arXiv preprint arXiv:2312.05247*, 2023. 2

[44] Zirui Wu, Tianyu Liu, Liyi Luo, Zhide Zhong, Jianteng Chen, Hongmin Xiao, Chao Hou, Haozhe Lou, Yuantao Chen, Runyi Yang, Yuxin Huang, Xiaoyu Ye, Zike Yan, Yongliang Shi, Yiyi Liao, and Hao Zhao. Mars: An instance-aware, modular and realistic simulator for autonomous driving. *CICAI*, 2023. 1, 2, 3

[45] Pengchuan Xiao, Zhenlei Shao, Steven Hao, Zishuo Zhang, Xiaolin Chai, Judy Jiao, Zesong Li, Jian Wu, Kai Sun, Kun Jiang, Yunlong Wang, and Diange Yang. Pandaset: Advanced sensor suite dataset for autonomous driving. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 3095–3101, 2021. 2

[46] Ziyang Xie, Junge Zhang, Wenye Li, Feihu Zhang, and Li Zhang. S-neRF: Neural radiance fields for street views. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 2

[47] Ze Yang, Yun Chen, Jingkang Wang, Sivabalan Manivasagam, Wei-Chiu Ma, Anqi Joyce Yang, and Raquel Urtasun. Unisim: A neural closed-loop sensor simulator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1389–1399, 2023. 1, 2, 3, 6

[48] Tianyuan Yuan, Yicheng Liu, Yue Wang, Yilun Wang, and Hang Zhao. Streammapnet: Streaming mapping network for vectorized online hd map construction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7356–7365, 2024. 2

[49] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 2

[50] Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, 34 (12):5586–5609, 2021. 3

[51] Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. *arXiv preprint arXiv:2312.07920*, 2023. 1