

---

# Supplementary Material for VR-Drive: Viewpoint-Robust End-to-End Driving with Feed-Forward 3D Gaussian Splatting

---

Hoonhee Cho<sup>1\*</sup> Jae-Young Kang<sup>1\*</sup> Giwon Lee<sup>1\*</sup> Hyemin Yang<sup>1\*</sup>  
Heejun Park<sup>1</sup> Seokwoo Jung<sup>2</sup> Kuk-Jin Yoon<sup>1</sup>  
<sup>1</sup> KAIST <sup>2</sup> 42dot

## A Dataset Licenses

In this work, we use the nuScenes dataset [1] which is published under the “Creative Commons license (CC BY-NC-SA 4.0)” and can be found under the URL <https://www.nuscenes.org/>. The carla simulator [2] is published under the “Creative Commons license (CC-BY)” and can be found under the URL <https://carla.org/>.

## B Implementation Details

We perform model training for 20 epochs on four A6000 GPUs, and conduct testing on a single A6000 GPU. The batch size is set to 4, and the learning rate is set to  $5e^{-5}$  with weight decay of 0.001 using AdamW optimizer. We use ResNet50 as image backbones and the image size of (256, 704). The confidence threshold  $\tau$  in Sec. 3.3 was set to 0.1. Our implementation primarily builds on modules from the following works [3, 4], except that the depth network is taken from [5] to ensure high quality.

**Anchor Boxes.** The anchor box is formulated as  $B \in \mathbb{R}^{M_d \times 11}$ , where  $M_d$  is the number of anchor boxes. Each anchor box is formatted with location, dimension, yaw angle and velocity:

$$\{x, y, z, \ln w, \ln h, \ln l, \sin yaw, \cos yaw, v_x, v_y, v_z\}.$$

**Anchor Maps.** The anchor map,  $L$ , is formulated as a polyline with 20 points:

$$\{x_0, y_0, x_1, y_1, \dots, x_{19}, y_{19}\}.$$

Then, anchor polylines  $L \in \mathbb{R}^{M_L \times 20 \times 2}$ , where  $M_L$  is the number of anchor polylines. The values of  $M_d$  and  $M_L$  were set to 900 and 100, respectively. The top- $K$  in the memory bank was set to 30 % of the number of anchors.

## C More Details about Sensor Setting

In real-world scenarios, an end-to-end autonomous driving system should be deployable across diverse vehicle platforms. To simulate sensor configurations for these varied vehicles, we represent camera extrinsics using three parameters: pitch  $\theta$ , height  $h$ , and depth  $d$ . Pitch denotes the vertical tilt angle of each camera, height is the vertical distance from the camera to the ground, and depth indicates the longitudinal offset from the vehicle center. By combining these parameters, we can simulate a wide range of real-world vehicle setups. As shown in the Fig. 1, each camera setting parameter represents the amount of change from the training setting. For example, after training on a

---

\*Equal contribution

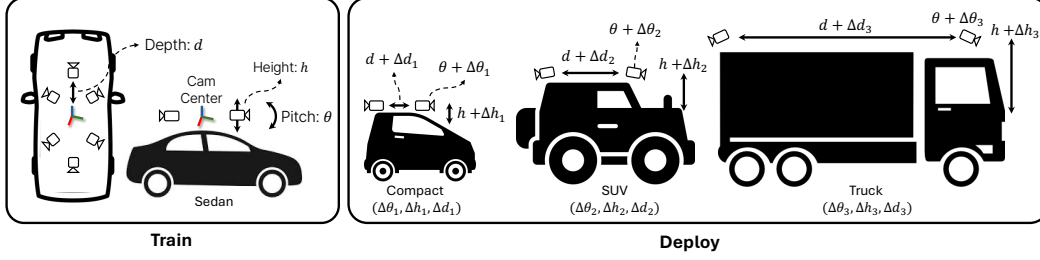


Figure 1: Camera Settings for Novel Viewpoints. To simulate novel camera viewpoints, we define three modulation parameters: pitch, height, and depth. Each parameter ( $\Delta\theta$ ,  $\Delta h$ ,  $\Delta d$ ) during evaluation represents the difference from the training settings. By adjusting these parameters, we can represent a wide range of realistic camera extrinsic setups.

Table 1: Hyper-parameter analysis of the confidence threshold,  $\tau$ . In the main paper, we set the  $\tau$  as 0.1.

$\tau$	Seen								Unseen Average							
	L2 (m) ↓				Collision (%) ↓				L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.
0.0	0.36	0.68	1.10	0.71	0.04	0.08	0.22	0.11	0.40	0.81	1.28	0.84	<b>0.01</b>	<b>0.07</b>	0.38	0.16
0.1	<b>0.29</b>	0.57	0.95	0.60	<b>0.01</b>	<b>0.03</b>	<b>0.14</b>	<b>0.06</b>	0.34	0.65	<b>1.06</b>	<b>0.68</b>	<b>0.01</b>	<b>0.07</b>	<b>0.24</b>	<b>0.11</b>
0.5	<b>0.29</b>	0.58	0.97	0.62	0.02	<b>0.03</b>	<b>0.14</b>	<b>0.06</b>	0.36	0.69	1.13	0.73	0.02	0.10	0.32	0.15
1.0	<b>0.29</b>	<b>0.56</b>	<b>0.93</b>	<b>0.59</b>	0.04	0.06	0.19	0.09	<b>0.33</b>	<b>0.64</b>	<b>1.06</b>	<b>0.68</b>	0.04	0.13	0.31	0.16

sedan setup, modulating pitch by  $-5^\circ$ , height by  $+1.0\text{m}$ , and depth by  $+1.0\text{m}$  simulates the camera viewpoint of a truck. By adjusting these parameters, we can effectively simulate a wide range of real-world vehicle setups.

## D Hyper-parameter Analysis

Table 1 provides an analysis of the hyper-parameters, including the distillation confidence threshold. The confidence threshold determines the selection of anchors that receive our proposed Viewpoint-Consistent Distillation. When the threshold is set to 0, distillation occurs for all anchors, which may include incorrect supervision from anchors that do not represent actual objects, leading to degraded performance. On the other hand, if the threshold is set too high, there will be fewer anchors eligible for cross-viewpoint distillation, making it less effective.

## References

- [1] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.
- [2] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017.
- [3] B. Liao, S. Chen, H. Yin, B. Jiang, C. Wang, S. Yan, X. Zhang, X. Li, Y. Zhang, Q. Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. *arXiv preprint arXiv:2411.15139*, 2024.
- [4] W. Sun, X. Lin, Y. Shi, C. Zhang, H. Wu, and S. Zheng. Sparsedrive: End-to-end autonomous driving via sparse scene representation. *arXiv preprint arXiv:2405.19620*, 2024.
- [5] Q. Tian, X. Tan, Y. Xie, and L. Ma. Drivingforward: Feed-forward 3d gaussian splatting for driving scene reconstruction from flexible surround-view input. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 7374–7382, 2025.