OccAny: Generalized Unconstrained Urban 3D Occupancy – Supplementary Material –

Anh-Quan Cao Tuan-Hung Vu
Valeo.ai, Paris, France
https://github.com/valeoai/OccAny

We provide in this document additional technical details in App. A, supplementary studies in App. B and qualitative examples in App. C.

A. Additional Details

A.1. Datasets

Occ3D-NuScenes was built upon nuScenes [2]. It contains $1,000$ 20-sec sequences captured by one LiDAR and six surrounding cameras. The dataset provides 3D occupancy annotations of 18 semantic classes, with 0.4 m voxels covering $80 \times 80 \times 6.4$ m areas at the resolution of $200 \times 200 \times 16$ voxels. Evaluation is done on the official *val* split [13] of 150 sequences.

SemanticKITTI, based on KITTI [1], consists of 22 sequences. Each sequence is annotated at the resolution of $256 \times 256 \times 32$ with 0.2 m voxels and 21 semantic classes (19 semantics, 1 free, 1 unknown). In our experiments, we only use images from the `cam2` camera. Following [3, 9], we evaluate on the *val* set, *i.e*. sequence 8.

A.2. Training

The *3D Reconstruction* stage (*cf*. Sec. 3.1) is trained in two consecutive steps:

- *Sequence-only training.* We only use mono-view sequences from all cameras across the five datasets. Training samples are drawn from frames within the same mono-view sequences.
- *Mixed training.* This step continues *Sequence-only training* while mixing surround-view data with sequential data (from the previous step) at a $1:1$ ratio. For surround-view data, we use frames from different cameras captured at the same timestep.

The *Novel-View Rendering* stage (*cf*. Sec. 3.2) is trained exclusively on sequential data. Empirically, we observed no gains when incorporating surround-view data in this stage.

Each stage is trained for 100 epochs using the AdamW optimizer [11] with a learning rate of 7×10^{-5}. We utilize a cosine scheduler with a minimum learning rate of 1×10^{-6} and a 3-epoch warmup. The training set consists of $50,000$ samples (sequences or sets of surrounding images), with $10,000$ drawn from each dataset. Experiments are conducted on 16 NVIDIA A100 40GB GPUs with an effective batch size of 64. The *3D Reconstruction* and *Novel-View Rendering* stages required approximately 40 and 30 training hours, respectively.

Method	Sem. feat.	Params	Semantic KITTI sequence			Occ3D-NuScenes surround-view		
			Res.	mIoU	mIoU^{sc}	Res.	mIoU	mIoU^{sc}
OccAny	Distilled	623M	512x160	**7.28**	**13.53**	512x288	6.66	10.32
OccAny+	Distilled	651M	512x160	6.48	13.30	512x288	**7.20**	**11.50**
OccAny	Pretrained	864M	512x160	7.67	13.75	512x288	7.42	10.78
OccAny+	Pretrained	1.08B	512x160	**8.03**	**13.17**	512x288	**9.45**	**12.22**

Table 6. **Using pretrained segmentation features** to boost semantic performance. OccAny+ is the variant using DA3 and SAM3 base models. Parameter counts reflect the forward path from the input to the predicted pointmaps and segmentation features. Note that using "pretrained" semantic features incurs a higher parameter cost due to the use of pretrained encoder.

A.3. OccAny+ using DA3 and SAM3

For the *3D Reconstruction* phase, we substitute the reconstruction encoder $\mathcal{E}$ and decoder $\mathcal{D}$ from MUSt3R with DA3 backbone, fine-tuning the final eight transformer layers and the dual DPT head. For novel view rendering, we utilize the same projection and tokenization layers but replace both the rendering encoder $\widetilde{\mathcal{E}}$ and decoder $\widetilde{\mathcal{D}}$ with DA3 backbone.

To leverage the strong initialization of the pretrained DA3 model, we introduce a self-distillation branch that duplicates the last eight transformer layers. These duplicated layers serve as a "teacher", supervising the eight trainable transformer layers via a scale-invariant loss, matching the scale-invariant output of the pretrained DA3.

In the *Novel View Rendering* phase, DA3 is initialized with the weights from the reconstruction phase. We train the first eight transformer layers while freezing the rest. Because DA3 lacks a memory mechanism, we tokenize the reconstruction outputs (pointmap, confidence, RGB, and

		Method	Extr.	Intr.	Fixed Ratio	Fixed Rig	GT LiDAR	Sem. Adapt.	GT Occ.	IoU	mIoU
Occ3D-NuScenes (ext. Tab. 3)	in-domain	SimpleOcc	Req.	Req.	Req.	Req.	Req.	Req.	–	33.92	7.05
		DistillNeRF	Req.	Req.	Req.	Req.	Req.	–	–	29.11	8.93
		SelfOcc	Req.	Req.	Req.	Req.	–	Req.	–	45.01	9.30
		POP-3D	Req.	Req.	Req.	Req.	Req.	Req.	–	28.17	9.31
		OccNeRF	Req.	Req.	Req.	Req.	–	Req.	–	39.20	9.53
		GaussianOcc	–	Req.	Req.	Req.	–	Req.	–	51.22	9.94
		VEON	Req.	Req.	Req.	Req.	Req.	Req.	Req.	57.92	12.38
		GaussTR	Req.	Req.	Req.	Req.	–	Req.	–	45.19	12.27
	out-of-domain	MUSt3R	–	–	–	–	–	–	–	13.61	2.43
		CUT3R*	–	–	–	–	–	–	–	19.21	3.06
		VGGT†	Rescale	Rescale	–	–	–	–	–	20.42	4.39
		AnySplat*†	Rescale	Rescale	–	–	–	–	–	20.78	4.44
		DA3	–	–	–	–	–	–	–	19.65	4.55
		OccAny	–	–	–	–	–	–	–	**34.10**	6.62
		OccAny+ (Pretrained)	–	–	–	–	–	–	–	33.49	**9.45**

*: use TTVA †: scaled with Metric3Dv2 [8].
Req.: required in-domain data/priors. Rescale: metric scaling needed

Table 7. **Detailed surround-view results.** OccAny+ is the variant using DA3 and SAM3 base models.

segmentation features) from all reconstructed views using the same tokenizer; these are passed alongside the novel-view tokens $\{\mathbf{X}_j\}_{j=1}^{N_{rnd}}$. To facilitate cross-view information exchange, we modify the attention mechanism to alternate between global and local attention starting from the first layer, rather than the eighth layer as in the original DA3.

Regarding *Segmentation Forcing*, we replaced the SAM2 encoder with a SAM3 encoder. Our findings indicate that performance improves significantly when the linear head is replaced with a DPTHead, particularly when trained using a $10\times$ higher learning rate. In all experiments, we use the DA3-LARGE variant.

B. Supplementary Studies

We present here the supplementary studies not presented in the main text due to the lack of space.

B.1. Boosting semantic performance.

While the unified OccAny model conveniently uses distilled segmentation features, it can also be combined with the original features from segmentation foundation models at inference. Although this introduces additional overhead, it enables the use of higher-resolution segmentation features and improves semantic performance, as shown in Tab. 6.

B.2. More surround-view results

Tab. 7 details results and method constraints in the surround-view setting, further including POP-3D [15], GaussianOcc [5], and VEON [20]. Existing in-domain approaches, including self-supervised ones, rely heavily on domain-specific priors, and VEON further depends on binary occupancy ground truth for training. In contrast, OccAny promotes a paradigm shift toward generalized and unconstrained occupancy prediction, enabling deployment of a *unified model* across out-of-domain and heterogeneous sensor setups. Beyond being unconstrained, OccAny can benefit from continual advances in foundation models, and is therefore expected to progressively narrow the remaining performance gap.

As preliminary evidence, upgrading MUSt3R to DA3 and replacing SAM2 with the more recent SAM3 yields an mIoU improvement of approximately 3 points, reaching performance comparable to recent self-supervised methods such as GaussianOcc [5].

B.3. Novel-View Rendering *vs*. Depth Completion

In this experiment, we compare the effectiveness of our *Novel-View Rendering* stage (*cf*. Sec. 3.2) with a baseline that performs depth completion on the projected pointmaps of the novel views. To this end, we replace *Novel-View Rendering* by using Prior Depth Anything [17], which takes as input the sparse projected pointmaps and the rendered RGB images produced by the state-of-the-art novel-view synthesis method AnySplat [10]. The Prior Depth Anything model outputs dense, completed depth maps for the novel views. We name this baseline $\text{OccAny}_{\text{depth completion}}$ and present comparison results in Tab. 8. Both models start from the first-stage-only OccAny and both adopt the TTVA strategy. OccAny significantly outperforms the $\text{OccAny}_{\text{depth completion}}$ baseline, validating the effectiveness of our second stage.

B.4. Generalization of State-of-the-art (SOTA) 3D Supervised Occupancy Models

We assess the generalization capability of SOTA 3D fully-supervised models by evaluating models *trained on a source dataset* directly on *a different target dataset*. We evaluate two settings:

- Occ3D-Waymo→ Occ3D-NuScenes (surround-view → surround-view).
- Occ3D-NuScenes/Occ3D-Waymo→ SemanticKITTI (surround-view→ monocular).

As shown in Table 9, despite careful alignment of sensor configurations, inference areas, and voxel resolutions, these supervised methods exhibit limited generalization capabilities compared to OccAny. Notably, OccAny's inference is straightforward and does not require any prior knowledge of the sensor configurations (number of cameras, intrinsics/extrinsics and camera poses), adapting effortlessly to any inference areas and any voxel resolutions.

Occ3D-Waymo→Occ3D-NuScenes. In this setting, we evaluate CVT-Occ [19] using weights trained on Occ3D-Waymo to perform inference on Occ3D-NuScenes. While the voxel resolutions and voxel sizes are consistent between these datasets, significant differences remain in sensor configurations. To enable inference, we align the sensor setups by mapping the five Occ3D-Waymo cameras to the six Occ3D-NuScenes cameras. Specifically, we map the

Method	Semantic KITTI							Occ3D-NuScenes						
	Res.	*sequence*			*monocular*			Res.	*sequence*			*surround-view*		
		Prec.	Rec.	IoU	Prec.	Rec.	IoU		Prec.	Rec.	IoU	Prec.	Rec.	IoU
OccAny$_{\text{depth completion}}$	512 × 160	24.59	44.55	18.82	21.59	**37.55**	15.89	512 × 288	29.80,	36.09	19.51	30.57	39.32	20.77
OccAny	512 × 160	**36.79**	**46.79**	**25.91**	**45.64**	33.66	**24.03**	512 × 288	**36.09**	**40.39**	**23.55**	**45.04**	**58.54**	**34.15**

Table 8. **Novel-View Rendering *vs*. Depth Completion.** Occupancy prediction results on SemanticKITTI and Occ3D-NuScenes show the effectiveness of *Novel-View Rendering*.

Label	Method	Venue	Occ3D-NuScenes surround-view				SemanticKITTI monocular			
			Res.	Prec.	Rec.	IoU	Res.	Prec.	Rec.	IoU
Occ	CVT-Occ [19] (Trained on Occ3D-Waymo)	ECCV'24	1600 × 900	35.38	25.86	17.56	1220 × 370	8.97	34.92	7.69
	CVT-Occ [19] (Trained on Occ3D-Waymo)	ECCV'24	960 × 540	29.15	28.33	16.78	960 × 292	8.92	36.84	7.73
	CVT-Occ [19] (Trained on Occ3D-NuScenes)	ECCV'24	in-domain	–	–	–	1220 × 370	11.73	**59.97**	9.43
	ALOcc [4] (Trained on Occ3D-NuScenes)	ICCV'25	in-domain	–	–	–	704 × 256	16.34	53.06	14.28
LiDAR	**OccAny**	–	512 × 288	**45.04**	**58.54**	**34.15**	512 × 160	**45.64**	33.66	**24.03**

Table 9. **Generalization results of fully-supervised methods.** *Occ* label is denser through temporal accumulation of LiDAR point-clouds and subsequent post-processing, whereas the *LiDAR* label remains sparser at each timestep. OccAny works out of the box in any evaluation settings with different inference areas, voxel resolutions and sensor configurations. In contrast, other methods require manual code modifications to align testing and training conditions. Beyond being more versatile, OccAny clearly demonstrates superior generalization.

Occ3D-Waymo Front, Front-Right, and Front-Left to their Occ3D-NuScenes counterparts, while the Occ3D-Waymo Side-Left is mapped to both Back and Back-Left, and Side-Right to Back-Right. Regarding image resolution, we follow the official implementation to scale Occ3D-NuScenes input images to the Occ3D-Waymo training resolution of 960×640. We also report inference performance at 1600×900, which yields slightly better results. However, as detailed in Table 9, even with these manual adaptations, the model struggles to generalize to the new domain, achieving a peak IoU of only 17.56%, significantly lower than the 34.15% achieved by our method.

Occ3D-NuScenes/Occ3D-Waymo→SemanticKITTI. Regarding the transfer from surround-view to monocular, we evaluate two SOTA 3D supervised methods: CVT-Occ [19] and ALOcc [4]. We use checkpoints trained on Occ3D-NuScenes (for both ALOcc and CVT-Occ) and Occ3D-Waymo (for CVT-Occ) to perform inference on SemanticKITTI. This scenario presents a significantly greater challenge than the previous setting: in addition to domain shifts and sensor discrepancies (using only the source front camera to align with the target setup), there are substantial divergences in voxel grid extents and resolutions.

For CVT-Occ [19], we use two provided models, one trained on Occ3D-NuScenes (1600×900) and another trained on Occ3D-Waymo (960×640). We evaluate the Occ3D-NuScenes-trained model on SemanticKITTI at full image resolution (1220×370), as it is closely aligned with the training resolution. For the Occ3D-Waymo-trained model, we conduct evaluations at both the full resolution and a resized resolution of 960×540, which preserves the SemanticKITTI aspect ratio while approximating the source training resolution.

For ALOcc [4], only the model trained on Occ3D-NuScenes is available. Since ALOcc encodes the stereo cost volume's frustum grid within its parameters, the network is constrained to a fixed input resolution of 704×256. Consequently, we evaluate ALOcc on SemanticKITTI at this exact resolution, adhering to the official implementation by using 16 history frames and pairs of consecutive timesteps as stereo input.

The results in Table 9 highlight a significant drop in performance when these models are inferred on the unseen SemanticKITTI dataset. CVT-Occ and ALOcc achieve IoUs of only 9.43% and 14.28%, respectively, whereas our proposed method demonstrates superior robustness with an IoU of 24.03%.

B.5. Ego Vehicle Trajectory Prediction

We assess the quality of ego-trajectory prediction using OccAny on the nuScenes validation set, following the evaluation protocol of [6, 7]. OccAny+ outperforms the base DA3-LARGE model in terms of Average Displacement Error (ADE), demonstrating clear advantages in urban scenes. Furthermore, it approaches the accuracy of optimization-based RGB-D SLAM methods while remaining fully feed-forward and significantly simpler.

Method	**ADE (m)**
GeoCalib [14] + DroidSLAM [12] + DA2 [18]	1.63
DA3 large + DA3 metric large	2.44
OccAny+	1.86

Table 10. **Ego Vehicle Trajectory Prediction.**

#Aug. Frames	1	2	4	8	10	20	50	100	200
Time (s)	0.052	0.057	0.085	0.172	0.227	0.542	1.406	2.786	5.572
Mem. (GB)	1.175	1.920	3.422	6.471	8.005	9.936	14.314	17.013	22.418

Table 11. **NVR inference complexity**, measured on one A100 GPU.

Method	**Train. GPUs**	**Train. time**	**Recon. time (ms)**	**Render time (ms)**	**Params (M)**
CUT3R	8×A100	≈30 days	240.0	259.8	793.3
VGGT	64×A100	>9 days	222.2	–	1157.9
AnySplat	16×A800	≈2 days	251.7	17.2	1190.7
OccAny	16×A100	≈1.5 days	93.8	123.2	651.1

Table 12. **Model size and speed.** Train times are from the original papers. Inference times are measured in the surround setting with 6 input views and 6 render views.

B.6. NVR complexity.

We report in Tab. 11 the memory consumption and running time of NVR inference using one A100 GPU in the surround-view setting. Similar to VGGT (*cf*. Tab. 9 in [16]), both memory & time scale much slower *w.r.t.* number of augmentation frames.

B.7. Model sizes and speeds

We are report the model sizes and speeds of OccAny and baselines in Tab. 12. OccAny has the fewest parameters (∼651M) *vs*. CUT3R (∼793M) and VGGT/AnySplat (∼1.2B), and is the most runtime efficient in training/inference. OccAny's rendering is about 2× faster than CUT3R, while AnySplat's is the fastest thanks to 3DGS.

C. Qualitative Examples

We show additional qualitative results in Fig. 9, Fig. 10, Fig. 11, Fig. 12, and Fig. 13.

References

[1] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Juergen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *ICCV*, 2019. 1

[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 *CVPR*, 2020. 1

[3] Anh-Quan Cao and Raoul de Charette. Scenerf: Self-supervised monocular 3d scene reconstruction with radiance fields. In *ICCV*, 2023. 1

[4] Dubing Chen, Jin Fang, Wencheng Han, Xinjing Cheng, Junbo Yin, Chenzhong Xu, Fahad Shahbaz Khan, and Jianbing Shen. Alocc: adaptive lifting-based 3d semantic occupancy and cost volume-based flow prediction. In *ICCV*, 2025. 3

[5] Wanshui Gan, Fang Liu, Hongbin Xu, Ningkai Mo, and Naoto Yokoya. Gaussianocc: Fully self-supervised and efficient 3d occupancy estimation with gaussian splatting. In *ICCV*, 2025. 2

[6] Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability. In *NeurIPS*, 2024. 3

[7] Mariam Hassan, Sebastian Stapf, Ahmad Rahimi, Pedro M. B. Rezende, Yasaman Haghighi, David Brüggemann, Isinsu Katircioglu, Lin Zhang, Xiaoran Chen, Suman Saha, Marco Cannici, Elie Aljalbout, Botao Ye, Xi Wang, Aram Davtyan, Mathieu Salzmann, Davide Scaramuzza, Marc Pollefeys, Paolo Favaro, and Alexandre Alahi. Gem: A generalizable ego-vision multimodal world model for fine-grained ego-motion, object dynamics, and scene composition control. In *CVPR*, 2025. 3

[8] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE TPAMI*, 2024. 2

[9] Yuanhui Huang, Wenzhao Zheng, Borui Zhang, Jie Zhou, and Jiwen Lu. Selfocc: Self-supervised vision-based 3d occupancy prediction. In *CVPR*, 2024. 1

[10] Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, Dahua Lin, and Bo Dai. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *ACM TOG*, 2025. 2

[11] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 1

[12] Zachary Teed and Jia Deng. DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras. In *NeurIPS*, 2021. 4

[13] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. In *NeurIPS*, 2023. 1

[14] Alexander Veicht, Paul-Edouard Sarlin, Philipp Lindenberger, and Marc Pollefeys. GeoCalib: Single-image Calibration with Geometric Optimization. In *ECCV*, 2024. 4

[15] Antonin Vobecky, Oriane Siméoni, David Hurych, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, and Josef Sivic. Pop-3d: Open-vocabulary 3d occupancy prediction from images. In *NeurIPS*, 2023. 2

[16] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, 2025. 4

[17] Zehan Wang, Siyu Chen, Lihe Yang, Jialei Wang, Ziang Zhang, Hengshuang Zhao, and Zhou Zhao. Depth anything with any prior. In *arXiv*, 2025. 2

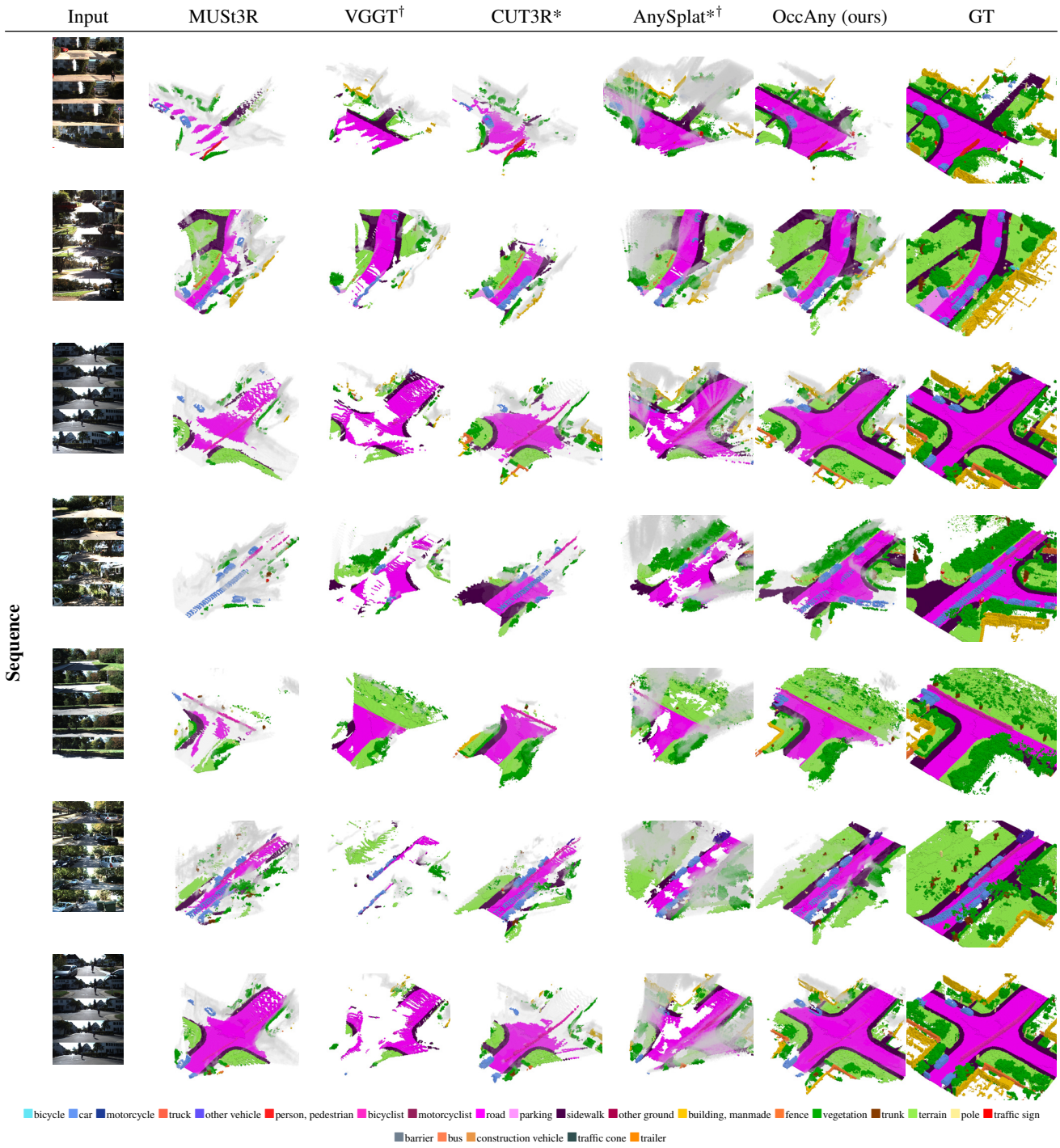


Figure 9. **Occupancy predictions** of OccAny and baselines on sequential data. We visualize here predicted voxels. For qualitative analysis, we overlay the semantic ground-truth colors on predicted voxels to better highlight class-wise gains. False positive voxels are painted in gray without any overlayed color. Compared to baselines, our occupancy predictions are denser and more accurate.

[18] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. In *NeurIPS*, 2024. 4

[19] Zhangchen Ye, Tao Jiang, Chenfeng Xu, Yiming Li, and Hang Zhao. Cvt-occ: Cost volume temporal fusion for 3d occupancy prediction. In *ECCV*, 2024. 2, 3

[20] Jilai Zheng, Pin Tang, Zhongdao Wang, Guoqing Wang, Xiangxuan Ren, Bailan Feng, and Chao Ma. Veon: Vocabulary-

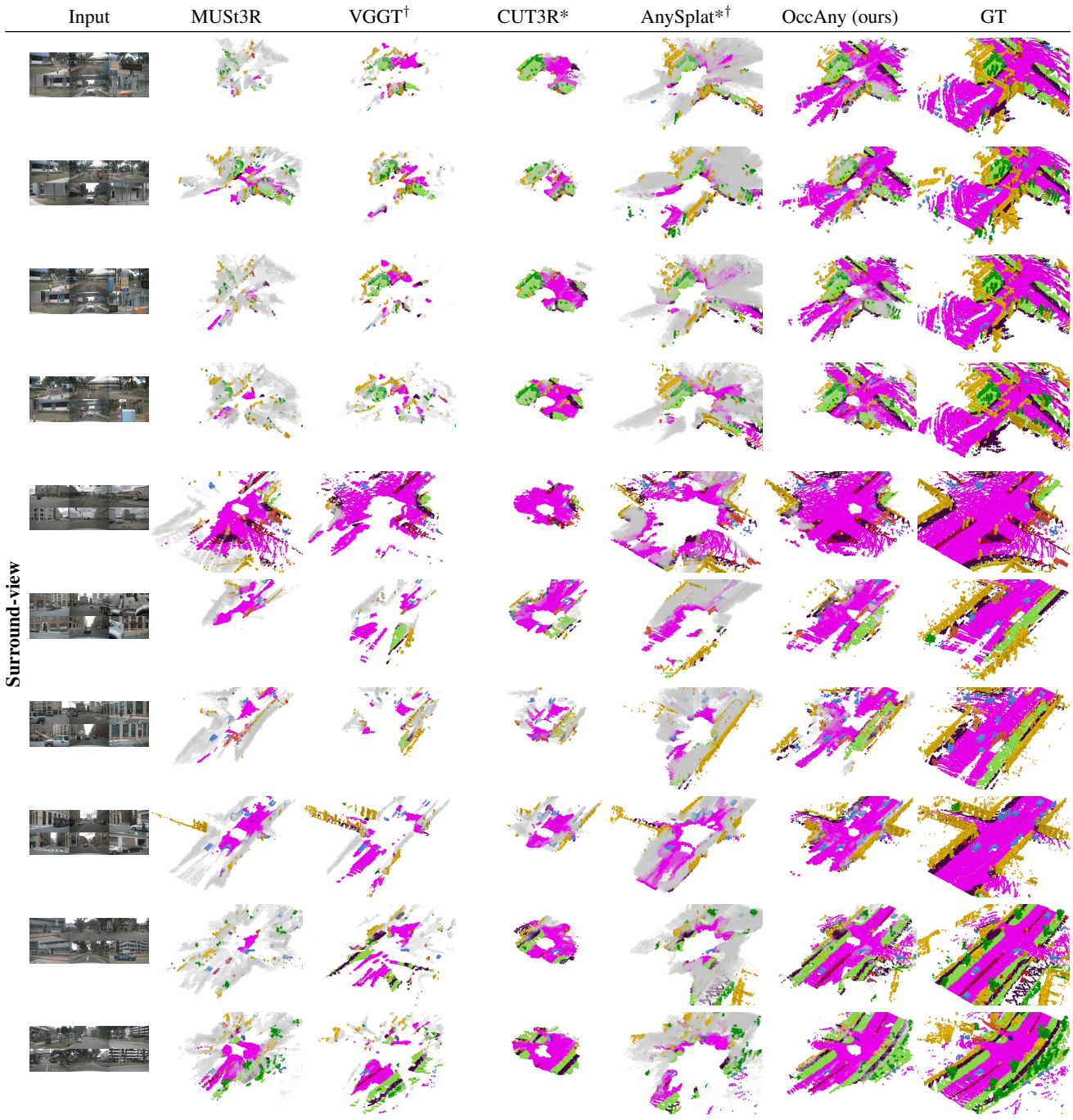
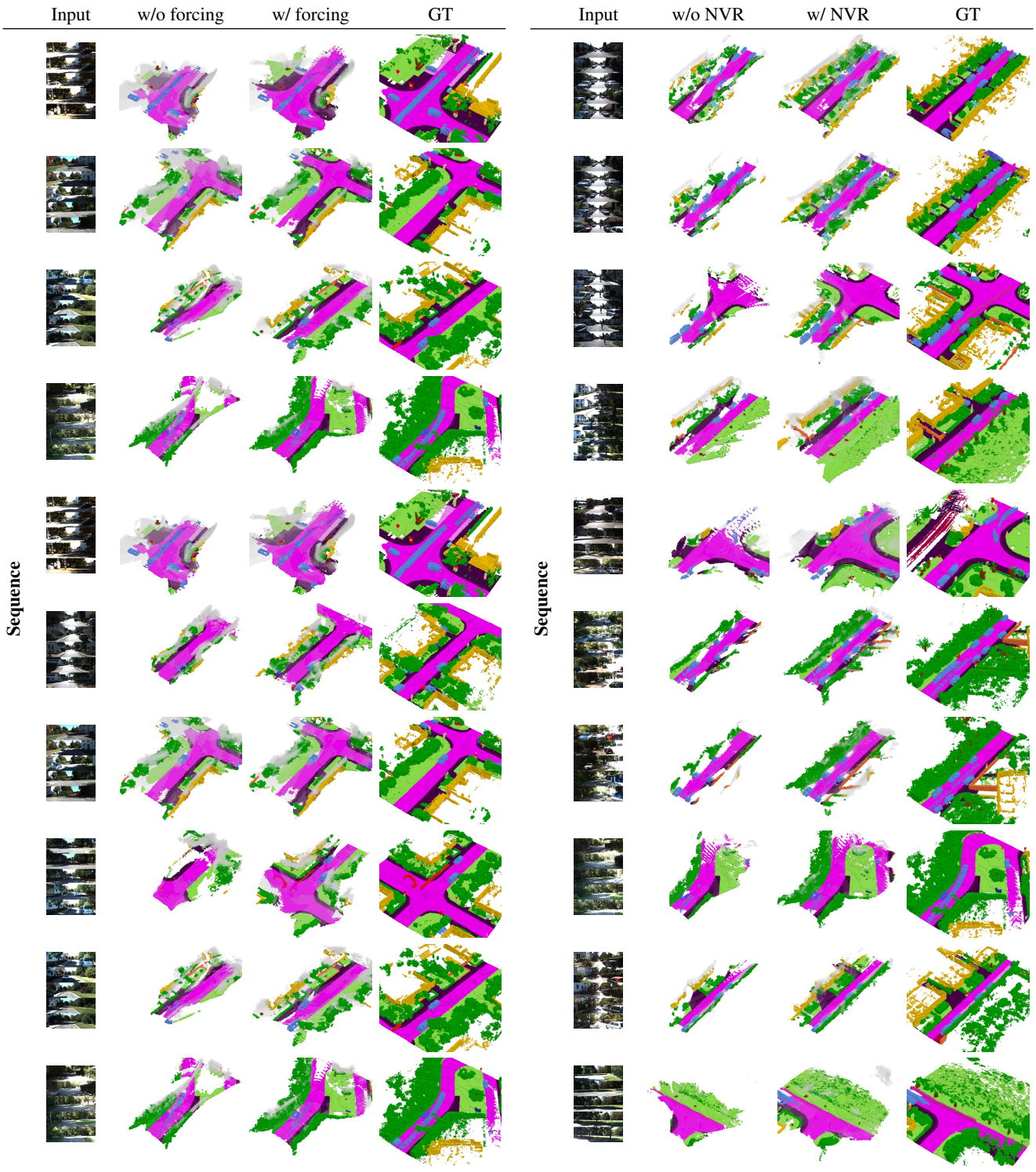


Figure 10. **Occupancy predictions** of OccAny and baselines on surround-view data. Voxel colorization follows Fig. 9. Compared to baselines, our occupancy predictions are denser and more accurate.

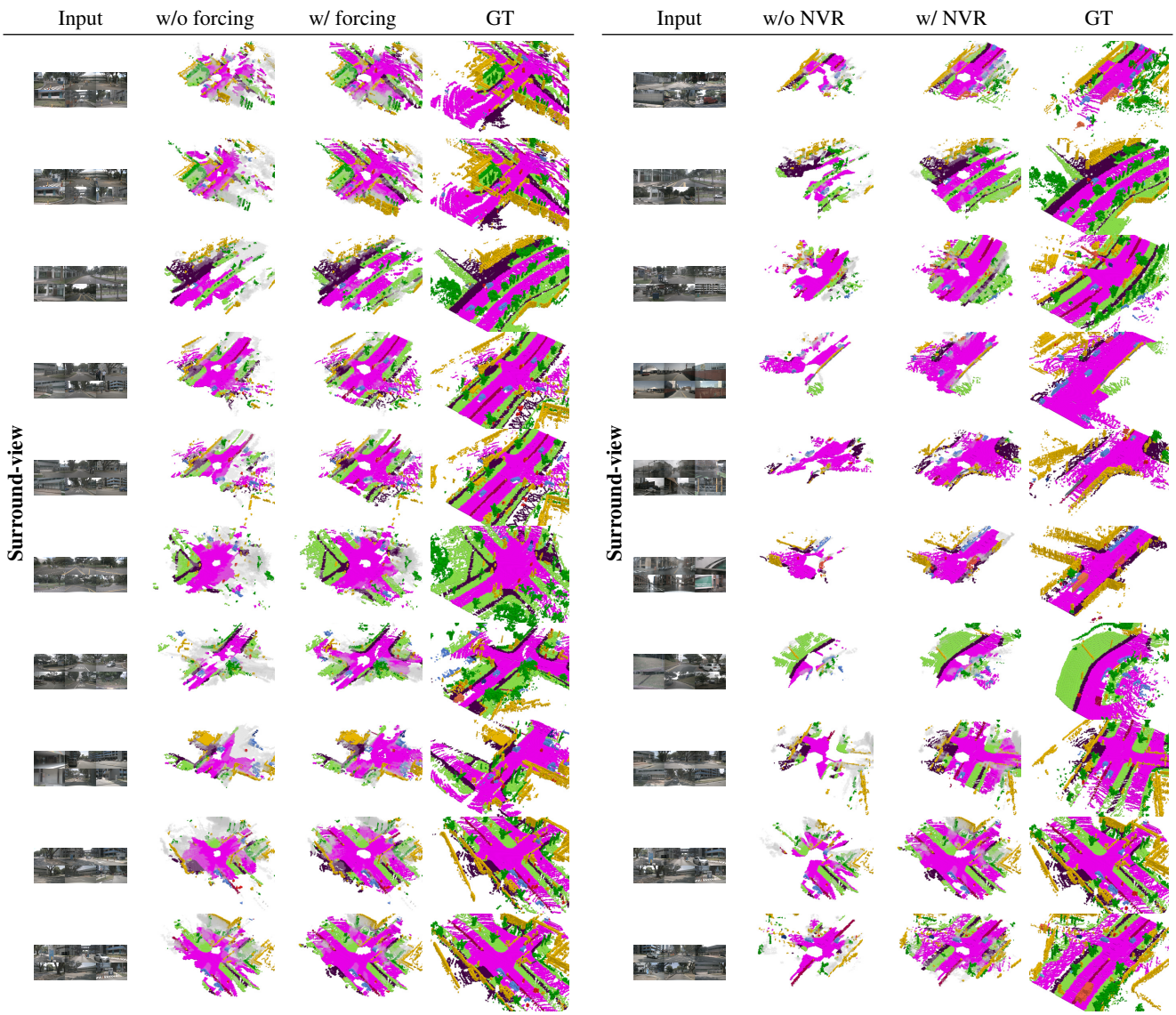
enhanced occupancy prediction. In *ECCV*, 2024. 2



(a) Segmentation Forcing

(b) Novel-View Rendering

Figure 11. **Qualitative ablation on Semantic KITTI** shows the gains from *Segmentation Forcing* and *Novel-View Rendering*. Voxel colorization follows Fig. 9. The two proposed strategies significantly improve the density and the accuracy of occupancy predictions.



(a) Segmentation Forcing

(b) Novel-View Rendering

Figure 12. **Qualitative ablation on Occ3D-NuScenes** shows the gains from *Segmentation Forcing* and *Novel-View Rendering*. Voxel colorization follows Fig. 9. The two proposed strategies significantly improve the density and the accuracy of occupancy predictions.

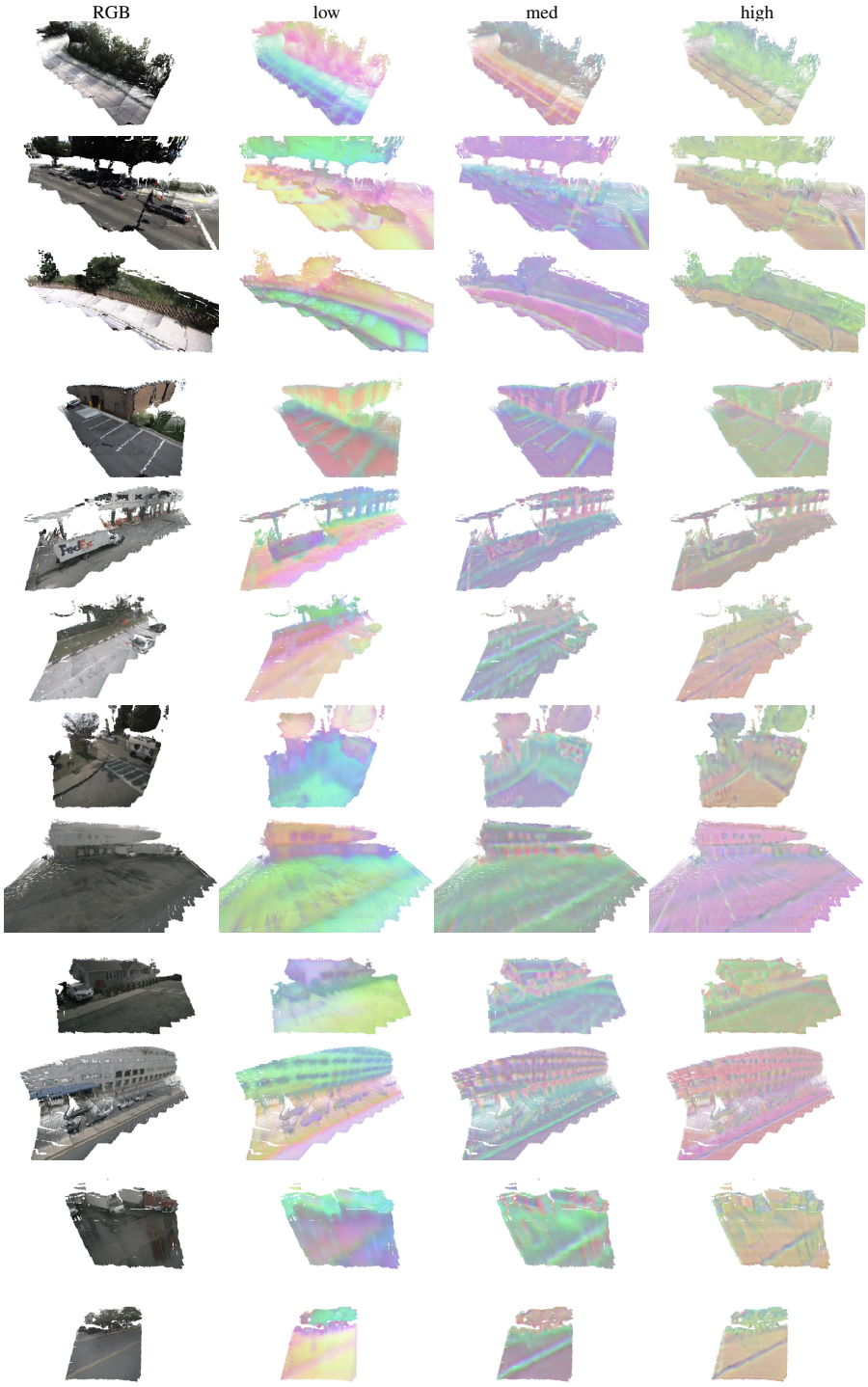


Figure 13. **PCA visualization of predicted feature maps.** Low-resolution features capture high-level semantics (e.g., separating cars, buildings, and roads), while high-resolution features capture low-level details such as boundaries and textures. Features remain consistent across different views.