

ViiNeuS: Volumetric Initialization for Implicit Neural Surface reconstruction of urban scenes with limited image overlap

Supplementary Material

In this supplementary material, we provide additional implementation details, experiments with Neuralangelo [6] and additional quantitative and qualitative results. Furthermore, an ablation study on our samples attribution strategy, applications, and failure cases of ViiNeuS are also provided.

1. SDF-gradient normalization

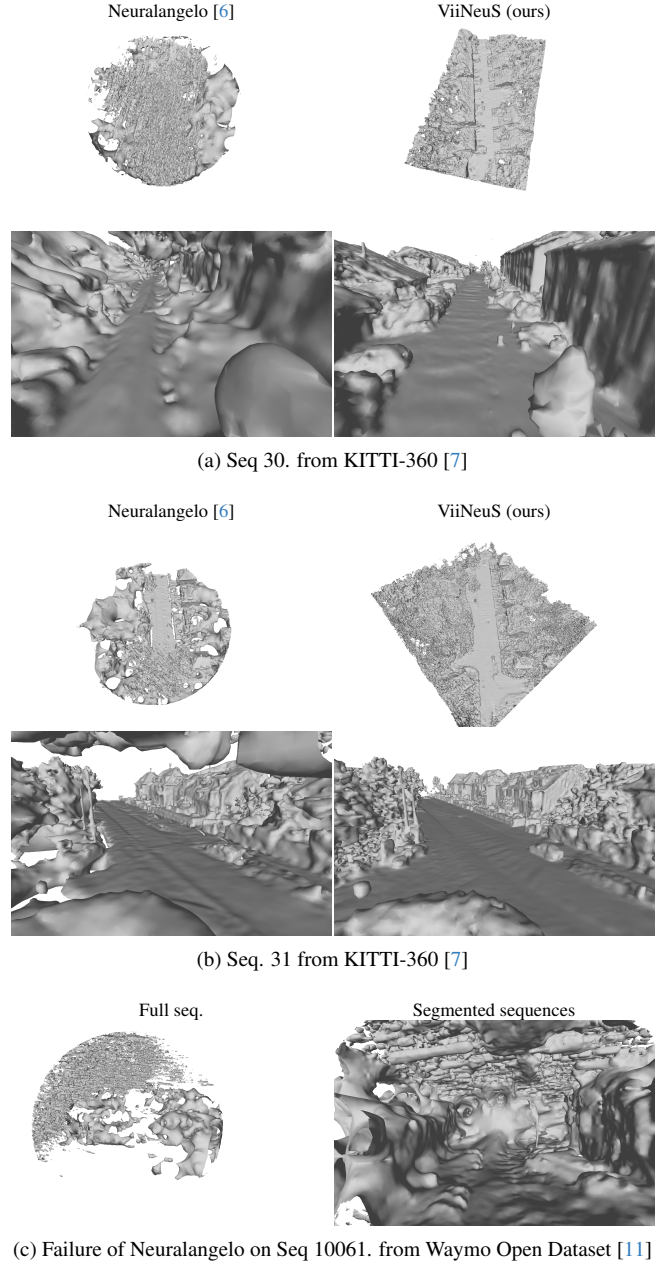
As explained in Sec. 3.3 of the main paper, volumetric representations based on density estimation tend to quickly approximate saturated alpha values (α_i^v being either 0 or 1), while alpha values computed from the SDF converge slower. Because α_i^v and α_i^f are composed jointly during the hybrid stage, we noticed that the surface representation compensates for this gap by predicting large SDF gradients so that α_i^f aligns with α_i^v . Indeed, from Eq. 3 of the main paper, a solution to saturate alpha towards either 1 or 0, is to predict $f(p_{i+1}) \ll f(p_i)$ or $f(p_{i+1}) \gg f(p_i)$, respectively. During the forward pass, $f(p_i)$ and $f(p_{i+1})$ are not directly predicted by the model but rather computed using:

$$\begin{aligned} f(p_i) &= f(x_i) + \text{Relu}(-\cos(\theta)) \times \frac{\delta_i}{2}, \\ f(p_{i+1}) &= f(x_i) - \text{Relu}(-\cos(\theta)) \times \frac{\delta_i}{2}, \end{aligned} \quad (1)$$

with θ being the angle between the direction of $\nabla f(x_i)$ and the ray direction d . Practically, we do not formally compute θ but we rather approximate its cosine with $\cos(\theta) = \nabla f(x_i) \cdot d$, **assuming both vectors are unit norm**. By predicting $\|\nabla f(x_i)\|_2 \gg 1$, α_i^f can be easily saturated and follow the distribution of α_i^v without learning a proper signed distance field. To address this, we simply normalize the SDF gradient before the cosine computation to prevent the gradient from compensating the alpha distribution difference. It is important to notice that even if the eikonal loss used for training (see main paper) is supposed to encourage the network to model a signed distance function with spatial derivative of unitary norm, we found that numerically normalizing the gradient during our hybrid stage is essential to avoid divergence in early training iterations.

2. SDF Field Initialization: Neuralangelo

As detailed in the main paper, Neuralangelo [6] is designed for landmark reconstruction, relying on many overlapping images and a bounded region of interest to initialize the SDF with a spherical shape. In our initial tests, when apply-



(c) Failure of Neuralangelo on Seq 10061. from Waymo Open Dataset [11]

Figure 1. Qualitative experiments results on Seq. 30 (a) and Seq. 31 (b) from KITTI-360s [7]. Failure on Seq. 10061 (c) Waymo Open Dataset [11]. We compare our generated SDF mesh to Neuralangelo [6] mesh for KITTI-360. We report the experiments on the full and segmented sequence for Waymo.

ing Neuralangelo (using the official authors codebase¹) to outdoor driving scenarios using the default settings for outdoor scenes, we observed that the initial part of the scene remained noisy and retained a spherical shape. Only the final part of the scene, which was consistently visible across all images in the sequence, was accurately reconstructed (see Fig. 1a). While Neuralangelo managed to reconstruct some parts of simpler scenes, such as in Seq. 31 from KITTI-360 (Fig. 1b), it failed completely in more complex cases, particularly in sequences that are long, wide, or contain challenging structure like downhills (e.g., Seq. 30 in Fig. 1a). Moreover, Neuralangelo was unable to reconstruct any part of the Waymo dataset, as shown in Fig. 1c. Given these outcomes, we conducted extensive experiments with Neuralangelo using various training strategies:

- Constraining the spherical shape: since Neuralangelo relies on a spherical initialization where everything outside the sphere is treated as background, we attempted to center the sphere on a smaller region, with a smaller radius. However, this approach resulted in very noisy reconstructions, likely due to insufficient overlapping images in that part of the scene (inside the sphere).
- Segmenting the scene into multiple parts: acknowledging that driving sequences are typically long and wide, we divided the scenes into smaller segments to better fit the spherical initialization. This strategy, however, produced unsatisfactory results. Neuralangelo requires a large number of images (typically 300 for an average Tanks and Temples scene covering the same region of interest) while KITTI-360 sequences contain around 200 images (approximately 50 per camera) and after segmentation, each part of the scene had roughly 100 images. The method could only reconstruct the final segment of the scene that was visible in all images (see Fig. 1c).
- Doubling the number of images: we further experimented with doubling the number of images used in the reconstruction. Unfortunately, this did not help the method to converge to a satisfactory reconstruction.

Despite various training strategies, Neuralangelo failed to produce reliable reconstructions for complex driving sequences. It requires a large number of overlapping images and a tightly bounded scene, making it unsuitable for unbounded driving sequences. Additionally, Neuralangelo’s training time is prohibitively long, taking up to 24 hours per scene. This, combined with its poor performance on driving sequences, makes it impractical for large-scale driving scene reconstruction tasks.

3. Comparison to feed-forward SfM solutions

To complete our extensive evaluation, we evaluated a feed-forward SfM solution designed for joint pose estimation

¹<https://github.com/NVlabs/neuralangelo>

Table 1. Quantitative evaluation results of Dust3r [12] on Waymo dataset.

	Dust3r	ViiNeuS
Seq. 10061	1.13	0.22
Seq. 13196	0.93	0.29
Seq. 14869	0.98	0.17
Seq. 102751	0.92	0.23

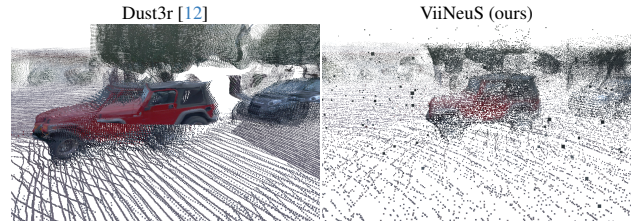


Figure 2. Point-cloud Seq. 13196 from Waymo.

and dense scene reconstruction rather than surface reconstruction. This method does not generate scene meshes or rely on precomputed poses for geometry estimation. We tested DUST3R [12] using prior poses and camera intrinsics to obtain a dense point cloud. DUST3R required 24GB of memory for depth map alignment with 60 downsampled images, necessitating a split into five chunks per sequence. Its output contained poor-quality results with duplicate content (qualitative comparisons in Fig. 2). We report the Chamfer distance between LiDAR and the predicted point cloud for DUST3R and ViiNeuS in Tab. 1.

4. Additional implementation details

We use the poses provided by the datasets, except for Waymo, where we recompute the vehicle trajectory and sensor calibration with MOISST [5] due to inaccuracies in the provided data. The overall loss we use to optimize ViiNeuS is defined as follows:

$$\mathcal{L} = \mathcal{L}_{\text{rgb}} + \lambda_1 \mathcal{L}_{\text{dssim}} + \lambda_2 \mathcal{L}_{\hat{N}} + \lambda_3 \mathcal{L}_{\text{eik}} + \lambda_4 \mathcal{L}_{\text{sky}}. \quad (2)$$

We set λ_1 , and λ_4 to 0.1 and 0.01, respectively. We fix λ_2 to 0.05 for planar classes and 0.01 for non planar classes. We set λ_3 to 0.01 in the first training iterations, then we adjust it to 0.1 in the last iterations.

We report in table 2 the hash grid encoding parameters from Instant-NGP [9]. We summarize the split of KITTI-360 [7] sequences used for our evaluations in table 3. We use all four cameras for KITTI-360 [7], and the three front cameras for Pandaset [13], nuScenes [1] and Waymo Open Dataset [11]. We sample one image out of two for KITTI-360, and one image out of 8 for the other datasets.

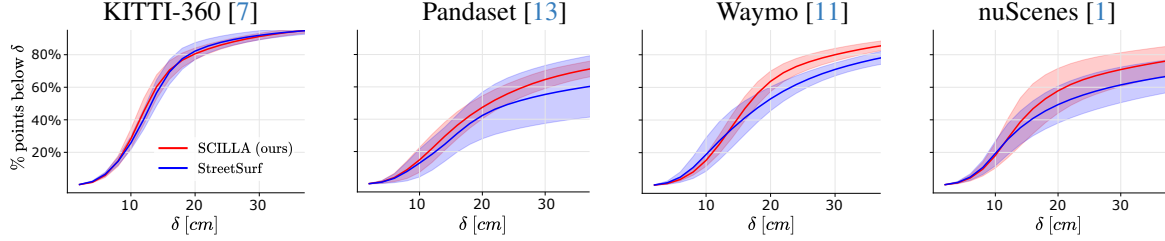


Figure 3. Mean cumulative delta error for both **ViiNeuS** and **StreetSurf** computed across the four sequences from each dataset as detailed in Tab. 1 of the main paper. Curves indicate the percentage of ground-truth points having an error distance to the closest predicted mesh triangle which is lower than a given value. Light contours represent the standard deviation for each method.

Table 2. Hash grid encoding parameters

Parameter	Value
Table size	2^{19}
Finest resolution	2048
Coarsest resolution	16
Number of level	16

Table 3. Selected KITTI-360 sequences

Seq.	KITTI Sync.	Start	End	# frames per cam.
30	0004	1728	1822	48
31	0009	2890	2996	54
35	0009	980	1092	57
36	0010	112	166	28

Table 4. Mean photometric results for each dataset

	KITTI		Pandaset		nuScenes		Waymo	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
StreetSurf	24.04	0.83	22.24	0.66	22.28	0.76	23.42	0.77
GOF	23.34	0.86	25.54	0.87	20.92	0.81	19.13	0.76
ViiNeuS	24.83	0.89	22.95	0.80	21.96	0.83	23.74	0.87

5. Additional results

5.1. Photometric results

We report the mean PSNR and SSIM for each dataset in Tab. 4. Our method shows comparable results to StreetSurf and GOF.

5.2. Additional quantitative results

We report in Fig 3 mean cumulative delta error and standard deviation computed across the four sequences from each dataset for both SDF methods. As it can be observed, our method’s cumulative errors are consistently lower in all datasets for distances below 40cm. We additionally show the standard deviation of the error computed along all sequences and notice that our errors remain consistent across the different scenes in contrast to StreetSurf.

5.3. Additional qualitative results

We report in Fig.4 additional qualitative results on nuScenes [1] and Waymo Open Dataset [11]. We find that ViiNeuS reconstructs higher-quality surfaces compared to StreetSurf and can recover many scene details (see highlighted red-boxes on the figures).

5.4. Ablation study

We ablate the effect of random sample attribution and our proposed probability-based samples attribution. The qualitative results at various training steps are presented in Fig.5. The results demonstrate that ViiNeuS samples attribution strategy initially learns the coarse geometry of the scene during early training stages. While random sample attribution can approximate an accurate SDF representation by the end of the hybrid stage, it results in an incomplete mesh compared to the mesh generated using probability-based sample attribution.

6. Applications

6.1. Textured mesh

Due to the high-quality of ViiNeuS’s reconstructed surfaces, we can leverage modern Multi-View Stereo (MVS) tools like OpenMVS [8] to produce detailed and colorized representations of driving sequences. As shown in Fig. 6, we find that ViiNeuS’s textured mesh is more complete and accurate compared to StreetSurf’s textured mesh.

7. Limitations

Although ViiNeuS’s reconstructed surfaces are highly detailed and accurate, we find that our method can fail in three distinct scenarios:

- Disentangling fine details from the sky: unlike StreetSurf [4], which models close range, far range, and sky separately, ViiNeuS separates only the sky from the other scene’s modeling. However, ViiNeuS may struggle to distinguish fine details from the sky, particularly in cases where objects are thin, as shown in Fig. 7.

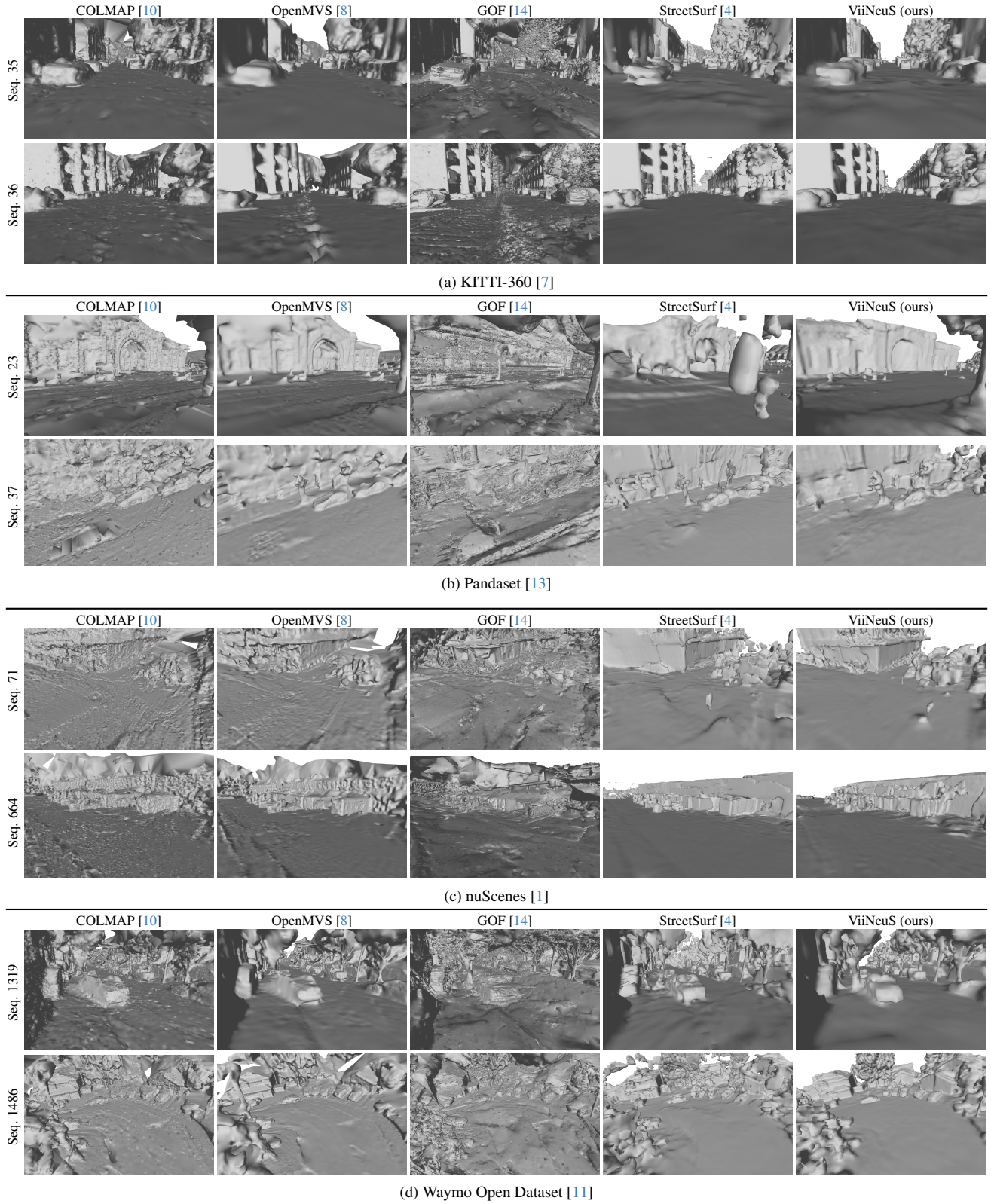


Figure 4. Qualitative experiments results on (a) KITTI-360s [7], (b) Pandaset [13], (c) nuScenes [1] and (d) Waymo Open Dataset [11]. We compare our mesh extracted from our SDF to GOF, COLMAP, OpenMVS and StreetSurf meshes.

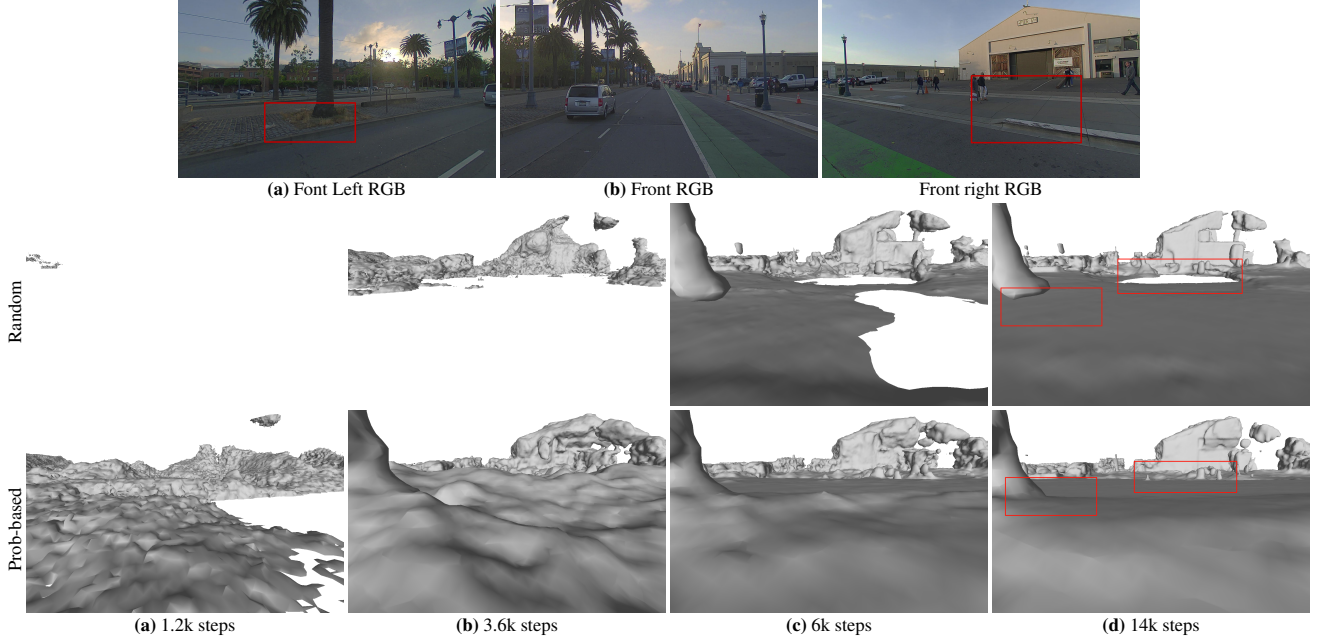


Figure 5. Ablation study: we ablate the effect of random samples attribution compared to our probability-guided attribution introduced in section 3.3 of the paper. We show the rendered meshes results of the sequence 023 of Pandaset [13] at (a) 1.2k steps, (b) 3.6k steps, (c) 6k steps, and (d) 14k steps.

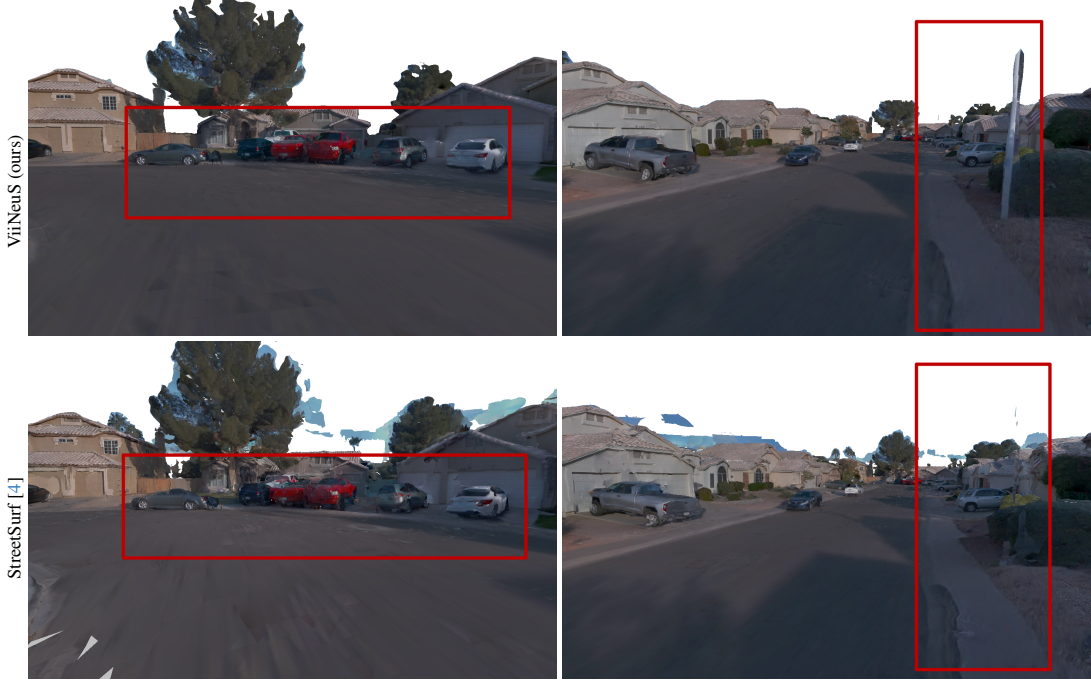


Figure 6. We use OpenMVS [8] to assign texture to the outputted meshes. We compare our colored mesh to StreetSurf [4], for the sequence 102751 from Waymo Open Dataset [11].

- Fine details in wide sequences: ViiNeuS sometimes fails in accurately reconstructing scene details in wide and open sequences, such as scene 23 from Pandaset reported in Fig. 8.
- Inaccurate road reconstruction: for sequence 664 from nuScenes (see Fig. 9), ViiNeuS faces challenges in re-

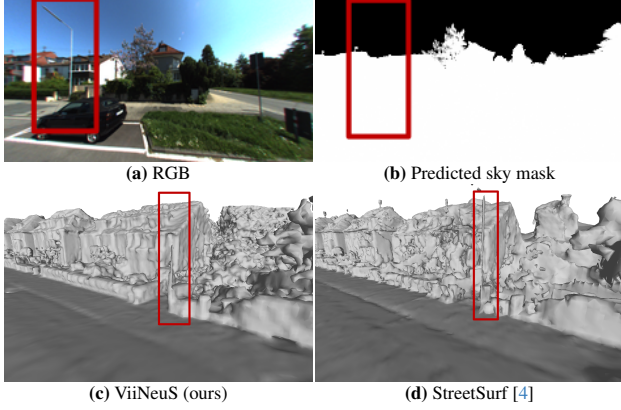


Figure 7. Failure case of ViiNeuS for the sequence 31 from KITTI-360 [7]. We report the ground-truth RGB image and the predicted sky mask. In a different point-of-view than the GT RGB, we compare our generated SDF mesh to StreetSurf’s mesh.

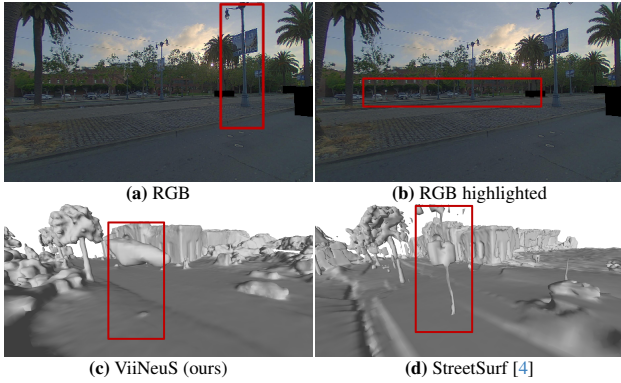


Figure 8. Failure case of ViiNeuS for the sequence 23 from Pandaset [13]. We report the ground-truth RGB image and the wide part of the sequence highlighted. In a different point-of-view than the GT RGB, we compare our generated SDF mesh to StreetSurf’s mesh.

constructing the road due to inaccuracies in the monocular normal prediction from Omnidata [3]. In comparison, StreetSurf [4] demonstrates more accurate road reconstruction, attributed to its road-surface initialization.

8. Supplementary video

We show in the supplementary video ViiNeuS’s meshes compared to GOF [14] and StreetSurf [4] on one sequence from each of the four evaluated datasets. All meshes were visualized with Blender [2] by animating the camera trajectory to generate the videos. GOF [14] meshes are incomplete at the beginning of scenes and very noisy, as the method is designed for landmark reconstruction and does not address the challenges of driving sequences, such as low image overlap, off-centered regions of interest, the need to handle both close and far-range objects across a wide range

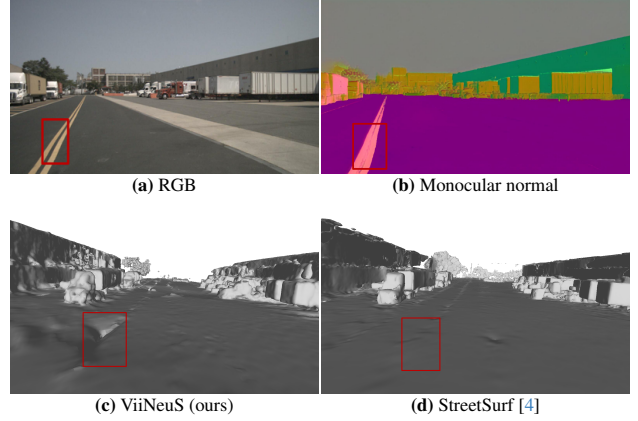


Figure 9. Failure case of ViiNeuS for the sequence 664 from nuScenes [1]. We report the ground-truth RGB image and the monocular normal. In a different point-of-view than the GT RGB we compare our generated SDF mesh to StreetSurf’s mesh.

of distances, and sky modeling. In addition, in the sky region and empty spaces that are commonly found in driving scenes, GOF tends to create triangles from noisy Gaussians. While the explicit 3DGS formulation is tailored for sparse scenes, it cannot effectively manage the inherent complexities introduced by the specific sensor configurations in driving sequences.

References

- [1] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nusenes: A multi-modal dataset for autonomous driving. In *CVPR*, 2020. [2](#), [3](#), [4](#), [6](#)
- [2] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. [6](#)
- [3] Ainaz Eftekhari, Alexander Sax, Jitendra Malik, and Amir Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *ICCV*, 2021. [6](#)
- [4] Jianfei Guo, Nianchen Deng, Xinyang Li, Yeqi Bai, Botian Shi, Chiyu Wang, Chenjing Ding, Dongliang Wang, and Yikang Li. Streetsurf: Extending multi-view implicit surface reconstruction to street views, 2023. [3](#), [4](#), [5](#), [6](#)
- [5] Quentin Herau, Nathan Piasco, Moussab Bennehar, Luis Roldão, Dzmitry Tsishkou, Cyrille Migniot, Pascal Vasseur, and Cédric Demonceaux. Moisst: Multimodal optimization of implicit scene for spatiotemporal calibration. In *IROS*, 2023. [2](#)
- [6] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *CVPR*, 2023. [1](#)
- [7] Yiyi Liao, Jun Xie, and Andreas Geiger. KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *PAMI*, 2022. [1](#), [2](#), [3](#), [4](#), [6](#)
- [8] Pierre Moulon, Pascal Monasse, Romuald Perrot, and Renaud Marlet. OpenMVG: Open multiple view geometry. In *International Workshop on Reproducible Research in Pattern Recognition*, 2016. [3](#), [4](#), [5](#)
- [9] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multi-resolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, 2022. [2](#)
- [10] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. [4](#)
- [11] Pei Sun, Henrik Kretschmar, Xerxes Dotiwala, 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 *CVPR*, 2020. [1](#), [2](#), [3](#), [4](#), [5](#)
- [12] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. [2](#)
- [13] Pengchuan Xiao, Zhenlei Shao, Steven Hao, Zishuo Zhang, Xiaolin Chai, Judy Jiao, Zesong Li, Jian Wu, Kai Sun, Kun Jiang, et al. Pandaset: Advanced sensor suite dataset for autonomous driving. In *ITSC*, 2021. [2](#), [3](#), [4](#), [5](#), [6](#)
- [14] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics*, 2024. [4](#), [6](#)