RoDUS: Robust Decomposition of Static and Dynamic Elements in Urban Scenes

Thang-Anh-Quan Nguyen[ID], Luis Roldão[ID], Nathan Piasco[ID],
Moussab Bennehar[ID], and Dzmitry Tsishkou[ID]

Noah's Ark, Huawei Paris Research Center, France
{thang.anh.nguyen,luis.roldao,nathan.piasco,
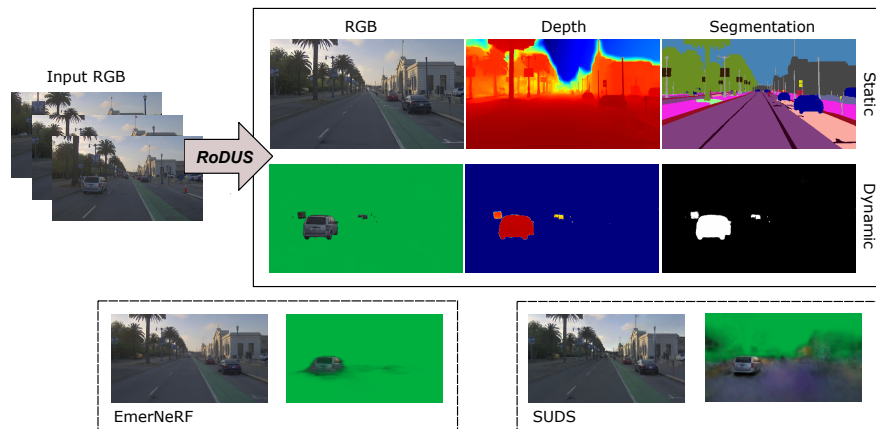moussab.bennehar,dzmitry.tsishkou}@huawei.com

Fig. 1: RoDUS is a neural scene representation designed to decompose 4D dynamic scenes into two elements: moving foreground and static background. This decomposition remains consistent across all photometric, geometric, and semantic aspects.

Abstract. The task of separating dynamic objects from static environments using NeRFs has been widely studied in recent years. However, capturing large-scale scenes still poses a challenge due to their complex geometric structures and unconstrained dynamics. Without the help of 3D motion cues, previous methods often require simplified setups with slow camera motion and only a few/single dynamic actors, leading to suboptimal solutions in most urban setups. To overcome such limitations, we present RoDUS, a pipeline for decomposing static and dynamic elements in urban scenes, with thoughtfully separated NeRF models for moving and non-moving components. Our approach utilizes a robust kernel-based initialization coupled with 4D semantic information to selectively guide the learning process. This strategy enables accurate capturing of the dynamics in the scene, resulting in reduced floating artifacts in the reconstructed background, all by using self-supervision. Notably,

experimental evaluations on KITTI-360 and Pandaset datasets demonstrate the effectiveness of our method in decomposing challenging urban scenes into precise static and dynamic components.

Keywords: Neural Radiance Fields (NeRFs) · Static-Dynamic Decomposition · Motion Segmentation

1   Introduction

There have been numerous works that achieved success in various 3D representations including both implicit [23] and explicit [36,37] scene modeling. Therefore, reconstructing and decomposing large-scale dynamic outdoor scenes is becoming a critical task for 3D scene understanding, especially for domains such as closed-loop simulation for autonomous driving [40,48,54]. Accurate segmentation and removal of dynamic objects are crucial as they are the main reason for artifacts during novel view synthesis (NVS) of background regions due to view inconsistency. This inevitably affects other downstream applications. For instance, several works [7,12,20] require masking out eventual dynamic objects as they estimate camera poses. Indeed, in highly dynamic scenes, without feeding the correct masks, they tend to make wrong refinements of the poses. On the other hand, dynamic objects often contain valuable information in the scene and play a main role in tasks such as surveillance and video compression [22].

However, advancements are lagging as understanding time-dependent scenes is an ill-posed problem. This is because it demands considerable efforts to detect and treat moving components separately while ensuring spatiotemporal consistency compared to static scenes [30,44]. The challenges further arise in driving settings, featuring not only large unbounded scenes, with lots of unconstrained dynamics and various illuminations but also large temporal gaps captured from the ego-vehicle. This demonstrates that a majority of the background regions are captured only for a short duration, due to high occlusions caused by other dynamic objects while the ego-vehicle is also in continuous motion. Unlike previous works [18,28,29,47], where scenes typically involve a single object with slow ego-motion, the scenarios encountered in driving sequences are less controllable.

Until now, the research community has proposed various solutions to the challenge of decoupling dynamic objects from static environments. Many develop NeRF-based multi-branch architectures in which only the dynamic (*i.e.* time-dependent) model is employed for rendering dynamic objects while the background is learned using a separate static model. Subsequently, these methods need to detect moving objects offline, so that the static model could later avoid sampling points at regions that belong to dynamic objects [15,27,48,50,51,53,60]. Nevertheless, they end up relying on expensive annotations for segmenting and tracking every object in the scene to enable correct guidance. Such limitation restricts their adaptability to many datasets. Others [4,21] learn to classify the inconsistency through a rejection factor while jointly optimizing both branches. However, NeRF's view-dependent capacity makes it difficult to distinguish non-

Lambertian but static from true dynamic objects, requiring careful and expensive parameter tuning to reach optimal separation.

In this paper, by building upon self-supervised 4D scene representation using NeRF, RoDUS (**Ro**bust **D**ecomposition of Static and Dynamic Elements in **U**rban **S**cenes) employs a two-pathway architecture with additional sky, road, and shadow modeling that is suitable for urban scenes. While the architecture has been mainly used for representing dynamic scenes as a whole, we argue that it appears a possibility of both branches learning incorrect information (*i.e.* local minima), as long as the overall reconstruction satisfies the ground truth image. This deviates from the initial design intent, where the static model is meant to learn only the static background, and vice versa. Toward this goal, we draw inspiration from [1] by proposing a robust kernel-based strategy, thus reducing L2 sensitivity on the static model. Additionally, we introduce a view-independent semantic field to prevent the dynamic model from expressing unwanted regions, while still being able to refine high-frequency detail. This semantic awareness is further used to guide the information flow, thus balancing the contribution of each branch. To summarize, the main contributions of our work are:

- We demonstrate RoDUS's capability of static-dynamic decomposition across photometric, geometric, and semantic outputs in a self-supervised manner.
- We introduce a robust initialization scheme and semantic reasoning ability with a masking mechanism that ensures the quality of separation.
- We conduct comprehensive comparisons on several challenging driving sequences, indicating our RoDUS's performance on static-dynamic components extraction, and decoupling compared to other SoTA methods.

2   Related Work

Recently, many works have achieved promising results in high-fidelity view synthesis [2] and combining with semantic [9,13,15,58], foundation model-guided [14, 41] reasoning to better accommodate urban driving scenarios [10,33]. As most of the above methods focus on static scenes, some other works have targeted non-static, transient elements from video sequences [17,18,28–30,44].

NeRFs for outliers rejection. NeRF-W [21] is the pioneering work that decomposes the scene into static and transient components to reconstruct landmarks from unconstrained phototourism datasets. They propose per-frame latent embeddings and a transient head to model different lighting conditions and transient effects, thus these phenomena can be removed from the rendering function at inference. Alternatively, RobustNeRF [34] successfully rejects distractors from the scenes using a trimmed kernel to modify the photometric loss into an Iteratively Least Squares problem, they also include patch filtering to maintain sharp reconstruction quality. Ha-NeRF [4], instead, parameterizes the kernel as learnable and uses 2D visibility map per image as a weight function. This occlusion-aware module is proven to be more accurate at separating transients.

Motion segmentation. In addition to advancements in 3D vision, there has been extensive research focusing on self-supervised motion segmentation at 2D

image level. The majority of these methods rely on 2D trackers [7], optical flow patterns [3,16,55] and/or image warping [35] to distinguish between ego motions and object motions and segment them. These approaches have clear limitations as they focus solely on image-level processing and lack scene-level understanding. Therefore, they struggle to handle complicated scenes with large camera motion [47]. Furthermore, those who follow the pipeline of *first-process-then-NeRF* create not only label ambiguity that affects learning but also add extra costs for separate perception processing tasks, considering that motion segmentation or flow classification themselves are already challenging.

NeRFs for dynamic scene reconstruction. To represent dynamic scenes, a direct solution is to add time as an additional input [17,32] or learn a deformation field that maps the input coordinates into a canonical space [28,29]. These time-dependent extensions, however, fail to accurately represent complex scenes and are unable to disentangle the dynamic components. Several works have adopted the idea in multi-branch architectures (*e.g.* NSFF [18], NeuralDiff [42], STaR [56], D^2NeRF [47]), decomposing the scene using different radiance fields. Nonetheless, the mentioned methods lack robustness and are not tested in urban driving scenarios where it is much harder to manage the number of dynamic objects within the scene. Consequently, their performance significantly degrades due to the blurriness of dynamic objects and floating artifacts in background regions, restricting their applicability. Follow-up works exploit object-centric approaches, especially NGS [27], PNF [15], and DisCoScene [50] express dynamic scenes and target automotive data, but they are fundamentally constrained by per-scene 3D annotations including bounding boxes and instance tracking which are difficult to obtain reliably without 3D sensors and proper calibration [11].

SUDS [43] and EmerNeRF [52] are the most similar to our approach since they utilize 2D off-the-shelf models to enable object-/scene-level semantic understanding while still being able to separate dynamic objects. However, our approach differs from existing works in that we place greater emphasis on the representation of each branch individually, a factor often overlooked by other studies. This focus proves advantageous in understanding complex scenes. Specifically, our dynamic model is designed to learn only the true dynamic instances rather than capturing variations in illumination or pixel changes caused by ego-motion.

3   Method

This section details RoDUS's methodology of disentangling and composing representation with its computational pipeline illustrated in Fig. 2. Sec. 3.1 describes the architecture for our scene representation. Sec. 3.2 introduces the semantic-radiance field used in RoDUS, how it can be rendered using differentiable volume rendering and contributes to the separation. Finally, we formulate the losses to train RoDUS in Sec. 3.3 as well as our proposed training strategy in Sec. 3.4.

Goals. Our main goal is to learn a scene representation of a dynamic environment (*e.g.* a driving scene) with the ability to semantically disentangle between static background (*e.g.* road, building, vegetation, parked vehicles) and dynamic

objects in a self-supervised manner. RoDUS takes a set of N input RGB images, their associated poses, and capturing timestamps $\{I_i, p_i, t_i\}_{i=1}^N$. Additionally, our model relies on semantic $\{S_i\}_{i=1}^N$, and geometric cues obtained from pre-trained 2D segmentation model and LiDAR depth measurements. At inference, we wish to achieve rendered images for static $\{\hat{I}_i^S, \hat{S}_i^S\}_{i=1}^N$ and dynamic $\{\hat{I}_i^D, \hat{S}_i^D\}_{i=1}^N$ components of the scene for each timestamp independently, which can also be combined for a complete view. As mentioned above, we study the quality of individual components rather than conforming solely to a satisfying composed result.

3.1   Scene Representation

Preliminaries. RoDUS is based on NeRF [23], which is designed to capture both geometry and view-dependent appearance of a scene from a set of RGB images. It encodes the scene within a Multi-Layer Perceptron (MLP) and outputs color $\mathbf{c}(\mathbf{x}, \mathbf{d}) \in \mathbb{R}^3$ and density $\sigma(\mathbf{x}) \in \mathbb{R}$ at any query position $\mathbf{x} \in \mathbb{R}^3$ along with its viewing direction $\mathbf{d} \in \mathbb{R}^3$. During training, using intrinsic parameters and poses, every pixel of the image undergoes a ray marching procedure $\mathbf{r}$ to sample inputs for the MLP. The model parameters are optimized to predict the colors rendered along the ray using the mean squared error: $\mathcal{L} = \|\hat{\mathbf{C}}_i(\mathbf{r}) - \mathbf{C}_i(\mathbf{r})\|_2^2$.

Architecture. Inspired by [18, 21, 47], we factorize the scene into two elements, each learned through a separate pathway. Each branch is modeled independently using its own multi-level hash grid representation and MLPs. This approach enables the concurrent learning of both aspects of the scene.

The static branch uses a 3D hash grid $\mathcal{H}^S$ for positions and positional embedding $\gamma(\cdot)$ for viewing directions, analogous to the one used in [25]. To model lighting variations, we also condition on a per-frame latent embedding l_a [21]:

$$(\sigma^S, \mathbf{c}^S, \mathbf{s}^S) = \mathcal{F}_\Theta^S(\mathcal{H}^S(\mathbf{x}), \gamma(\mathbf{d}), l_a). \tag{1}$$

As the dynamic branch allows its outputs to depend on time, we extend to 4D hash grid $\mathcal{H}^D$ with one additional normalized timestamp $t \in [0, 1]$ as input. We additionally incorporates a shadow head [47], designed to learn a shadow ratio $\rho \in [0, 1]$. This scalar is later used to scale down the static radiance values $\mathbf{c}^S$ in Eq. (3):

$$(\sigma^D, \mathbf{c}^D, \mathbf{s}^D) = \mathcal{F}_\Theta^D(\mathcal{H}^D(\mathbf{x}, t), \gamma(\mathbf{d})),$$
$$\rho = \mathcal{F}_\Theta^\rho(\mathcal{H}^D(\mathbf{x}, t)). \tag{2}$$

Rendering. We calculate the color $\hat{C}(\mathbf{r})$ of each ray $\mathbf{r}$ by integrating all the sampled points along the ray:

$$\hat{C}(\mathbf{r}) = \sum_{i=1}^K T_i \left(\alpha_i^S (1 - \rho_i) \mathbf{c}_i^S + \alpha_i^D \mathbf{c}_i^D \right), \tag{3}$$

where $T_i = \exp\left(-\sum_{j=1}^{i-1}(\sigma_j^S + \sigma_j^D)\delta_j\right)$ is the accumulated transmittance and $\alpha_i^\xi = 1 - \exp(-\sigma_i^\xi \delta_i)$ with $\xi \in \{S, D\}$ are alpha values. Given the outputs of each branch, we can also render static and dynamic maps separately.
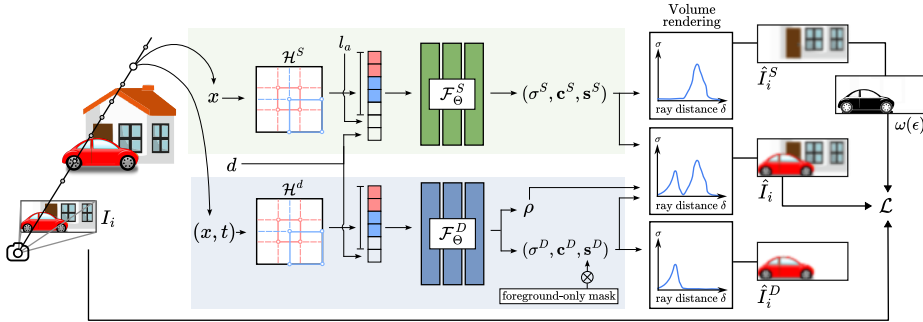
Fig. 2: RoDUS's architecture. Our model comprises two separate branches that take as input sampled positions $\mathbf{x}$, viewing direction $\mathbf{d}$, and their timestamp t, and generate outputs for every query coordinate. Each branch is represented by a separate hash grid and the MLP-based neural function, which predicts colors, densities, and semantics. The rendered static RGB is used to calculate IRLS map $\omega(\epsilon)$ during the robust initialization step (Sec. 3.3), while dynamic semantic outputs are passed through a "foreground-only mask" to prevent over-explaining background regions (Sec. 3.2).

3.2   Enabling Semantic Awareness

NeRF provides low-level implicit representations of geometry and radiance but lacks a higher-level understanding of the scene. It is shown in [34] that relying only on view-dependent radiance and sensitive mean square error makes the model prone to outliers and over-fit observations. On the other hand, semantic labels can be conceptualized as a *view-invariant* function [59], mapping world coordinates $\mathbf{x}$ to a distribution over semantic labels $\mathbf{s}(\mathbf{x}) \in \mathbb{R}^L$, where L is the number of classes. We choose to integrate a semantic head to predict the semantic class for each point, similar to the color head. Subsequently, the same rendering formula is applied for semantics to achieve 2D semantic segmentation maps:

$$\hat{S}(\mathbf{r}) = \sum_{i=1}^{K} T_i \left(\alpha_i^S \mathbf{s}_i^S + \alpha_i^D \mathbf{s}_i^D \right). \tag{4}$$

The use of semantic supervision in RoDUS serves two main purposes. First, it provides assumptions for multiple classes, forming the basis for the loss formulations (refer to Sec. 3.3) that contribute to enhancing rendering quality. Second, it enables the model to capture detailed scene layout and object boundary information with a high level of precision thereby assisting in decomposition.

One point to consider is that supervision for segmentation with noisy pseudo labels should not impact the underlying geometry. Otherwise, the model can still satisfy conflicting labels by changing the dynamic weights, resulting in the wrong geometry. Previous works [9, 39] stop gradients from the semantic head back to the densities to solve this problem. However, we believe that semantic information plays an important role in correcting the separation. We instead limit the dynamic model from contributing static class prediction to the final results
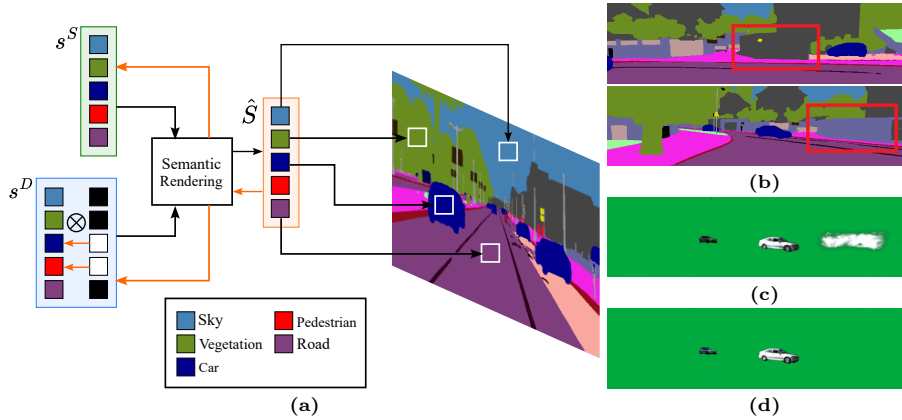
Fig. 3: (a) We enforce a "foreground-only mask" to the dynamic semantic head. In the forward pass (black arrows), the mask prevents the dynamic branch from outputting pixels that do not belong to foreground classes. While in the backward pass (orange arrows), it restricts the dynamic field from learning background pixels, which may primarily come from noisy annotations. (b) Predictions generated by 2D segmentation model are noisy and inconsistent between views. (c) Since the dynamic field includes a temporal dimension, it ends up learning these conflicting labels, satisfying the overall loss. (d) Therefore, our proposed mask is used to tackle the problem.

by putting a mask in the last layer of its semantic head. This masking strategy can be viewed as a form of gradient stopping the learning but in a selective way as illustrated in Fig. 3. Note that we do not have motion information for each instance, we only leverage the fact that certain classes are potentially movable to penalize the dynamic model from predicting background (non-movable) samples.

We apply `softmax` normalization at each query point individually before ray integration, as introduced in [39]. This is done due to the unbounded nature of logits, ensuring that the semantic head can not generate excessively high scores in regions with low density to overcome high-density areas. Remarkably, this not only improves the stability of rendering semantics but also prevents half-precision floating point (`float16`) overflowing when working with frameworks such as `tiny-cuda-nn` [24]. By performing inferences in 3D space and rendering to 2D with a foreground masking mechanism, our semantic outputs are multi-view consistent by design.

3.3   Loss Formulation

Reconstruction losses. Consistent with other NeRF works, we incorporate both L2 and DSSIM [46] losses to supervise the pixel color $\mathcal{L}_{rgb}$. In cases where LiDAR data is available, depth supervision [8,33] loss $\mathcal{L}_d = \|\hat{d}_i(\mathbf{r}) - d_i(\mathbf{r})\|_2^2$ is considered to enhance the reconstruction accuracy. In addition, cross-entropy [59] loss is applied to provide supervision on the generated semantic labels derived from pseudo-ground truth segmentation, expressed as $\mathcal{L}_{sem} = CE(\hat{S}(\mathbf{r}), S(\mathbf{r}))$.
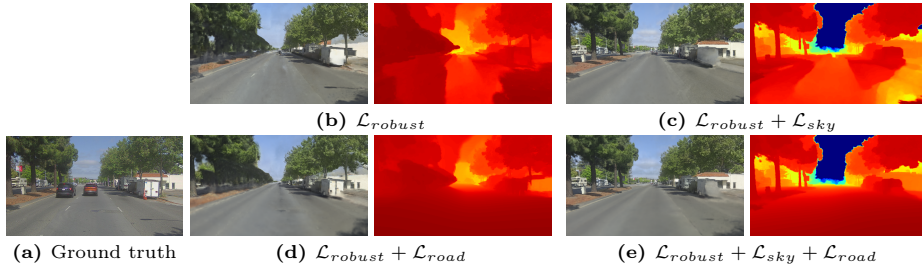
(b) $\mathcal{L}_{robust}$ **(c)** $\mathcal{L}_{robust} + \mathcal{L}_{sky}$

(a) Ground truth   **(d)** $\mathcal{L}_{robust} + \mathcal{L}_{road}$ **(e)** $\mathcal{L}_{robust} + \mathcal{L}_{sky} + \mathcal{L}_{road}$

Fig. 4: (b) Using a trimmed kernel effectively removes all moving cars from the scene. (c, d, e) As the sky and road regions are badly reconstructed, applying class constraints can aid the learning process of the static model.

Robust loss. Our objective is to reduce the sensitivity of the static model on dynamic components through a robust photometric loss. This loss can be formulated as an Iteratively Reweighted Least Squares (IRLS) problem:

$$\mathcal{L}_{robust} = \kappa \left(\|\hat{C}^S(\mathbf{r}) - C(\mathbf{r})\|_2 \right) = \omega(\epsilon(\mathbf{r})) \cdot \|\hat{C}^S(\mathbf{r}) - C(\mathbf{r})\|_2^2, \qquad (5)$$

where $\epsilon(\mathbf{r}) = \|\hat{C}^S(\mathbf{r}) - C(\mathbf{r})\|_2$ is the error residuals and $\omega(\epsilon) = \epsilon^{-1} \cdot \partial\kappa(\epsilon)/\partial\epsilon$ is the weight function of the residuals at the previous iteration. Among the family of kernels [1], we follow Sabour *et al.* [34] and choose a trimmed estimator [6] which classifies outliers based on a percentile threshold $\mathcal{T}_\epsilon$ calculated from the whole batch of rays. This choice ensures outliers are completely removed (*i.e.* hard weights). Furthermore, as every object has neighbor connections, we apply a spatial filter $\mathcal{F}$ on the residuals map to capture the smoothness, ensuring local support of the kernel:

$$\omega(\epsilon(\mathbf{r})) = \mathcal{F}\left[\epsilon(\mathbf{r}) \leq \mathcal{T}_\epsilon\right], \qquad (6)$$

where $\mathcal{F}$ consists of a box filter to diffuse inlier/outlier labels and a sub-patch level consistency filter that takes into account the rejection behavior of its surrounding neighborhoods. This filter aims to remove high-frequency details from being classified as outliers and allows them to be captured by the NeRF models.

Sky loss. Outdoor scenes contain sky regions where rays never intersect any opaque surfaces. We detect pixels that belong to the sky and force the densities to be zeros along the ray, excluding the last sample:

$$\mathcal{L}_{sky} = \mathbb{E}\left[\frac{1}{K-1}\sum_{i=1}^{K-1}\sigma_i + \frac{1}{\sigma_K}\right]_{\mathbf{r}\in\mathcal{R}_{sky}}. \qquad (7)$$

Planar regularization loss. We observe that the road surface is challenging to model as it has low textures and is often occluded by moving vehicles. These areas mostly suffer from poor geometry prediction (Fig. 4). To address this issue, we rely on the work of Wang *et al.* [45], perform Singular Values Decomposition
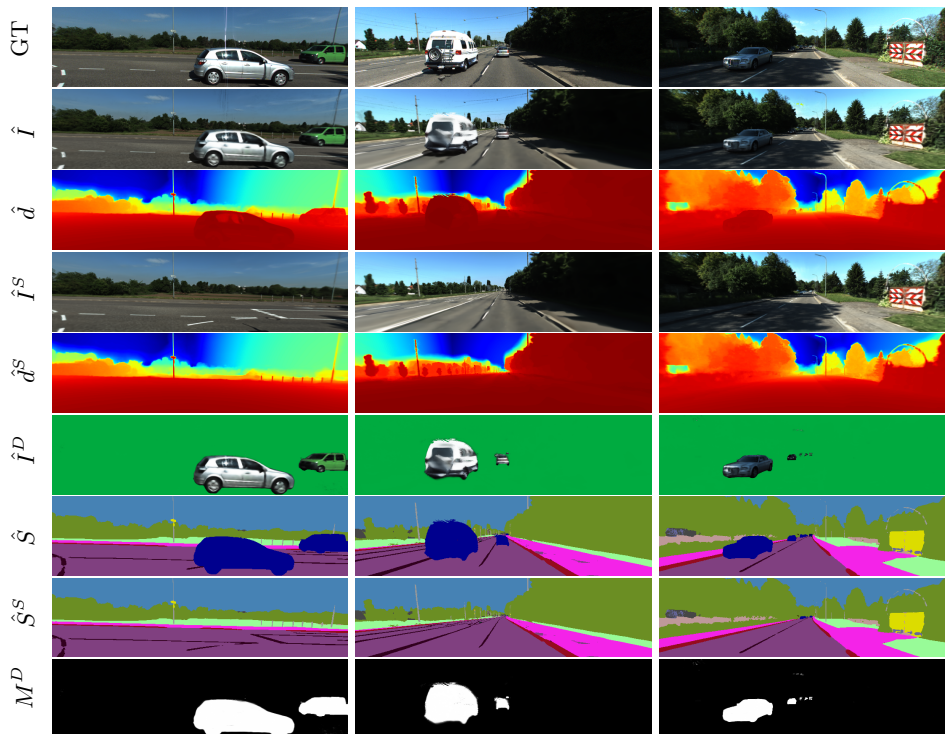
Fig. 5: We demonstrate RoDUS's capabilities on multiple tasks. *Legend:* GT: Ground truth, $\hat{I}$: Composed RGB, $\hat{d}$: Composed depth, I^S: Static RGB, $\hat{d}^S$: Static depth, $\hat{I}^D$: Dynamic RGB (blended with a green background), $\hat{S}$: Composed segmentation map, $\hat{S}^S$: Static segmentation map, M^D: Motion segmentation.

(SVD) on the set of centered points projected from patches that are assigned with the road label, and minimize the smallest singular value σ_3:

$$\mathcal{L}_{road} = \sigma_3(\mathcal{R}_{road}^{patch}). \tag{8}$$

Dynamic regularization loss. We use the factorization $\mathcal{L}_{\mathcal{H}}$ and sparsity $\mathcal{L}_{\sigma^D}, \mathcal{L}_{\rho}$ losses that have been proposed in [38, 47, 56] to enforce the static branch to explain most of the static structure as possible and only express the moving objects and/or shadows using the dynamic branch.

Our final loss results on the weighted sum of all our individual losses:

$$\mathcal{L}_{total} = \underbrace{\mathcal{L}_{rgb} + \lambda_d \mathcal{L}_d + \lambda_{sem} \mathcal{L}_{sem}}_{reconstruction}$$
$$+ \underbrace{\mathcal{L}_{robust} + \lambda_{sky} \mathcal{L}_{sky} + \lambda_{road} \mathcal{L}_{road}}_{static\ regularization} + \underbrace{\lambda_{\sigma^D} \mathcal{L}_{\sigma^D} + \lambda_{\rho} \mathcal{L}_{\rho} + \lambda_{\mathcal{H}} \mathcal{L}_{\mathcal{H}}}_{dynamic\ regularization} \tag{9}$$

3.4   Training Strategy

The use of robust estimation plays an important role in rejecting inconsistency (dynamic objects in our case) and guiding the static branch without any prior knowledge. However, we observe that when using a binary kernel alone with a fixed threshold, after reaching a certain point throughout the training, the robust loss does not contribute to the overall loss anymore. We refer to this phenomenon as "kernel saturation", this causes several parts of the scene not to be learned completely. Adjusting the threshold $\mathcal{T}_\epsilon$ seems to be a trivial solution, but given the fact that outdoor scenes are more complicated to handle, there is always a trade-off between the number of distractors removed and the static scene quality. Particularly, high-frequency but static details, equivalently appear as large residuals are considered as outliers, while low-frequency but moving regions such as shadows (**dark**) which mostly share a similar color with the road (gray) are uncontrollably included as inliers. Fig. 6 portrays this, as there are still parts of the scene (*e.g.* road marking) that are not fully learned.

In contrast, relying solely on reconstruction losses leads to several local optima during optimization due to the entanglement of geometry and appearance. We propose using the robust loss as an initialization step for the static model to learn the basic overall structures of the scene before delving into complete scene understanding. The initialization step guides the network to prioritize consistent (or confident) regions that further help stabilize the training process. Therefore, we start the model training with the robust loss (Eq. (5)) and sky loss (Eq. (7)) enabled for the first epoch, and subsequently enable the remaining losses. By doing so, we can confidently select a low threshold without compromising the quality. This serves as a foundation for disentangling the scene in later training stages.

4   Evaluation

4.1   Experimental Setup

Implementation details. We build RoDUS using PyTorch [31]. Our implementation is based on `tiny-cuda-nn` [24] from iNGP [25], and proposal sampling strategy inspired from MipNeRF360 [2]. To generate pseudo-ground truth for semantic segmentation, we use the Mask2Former [5] model pre-trained on the Mapillary Vistas [26] dataset.

Datasets. We evaluate our system on two challenging urban driving datasets, KITTI-360 [19] and Pandaset [49], both featuring dynamic objects in driving scenarios. In Pandaset, we utilize three front cameras and one back camera, while in KITTI-360, we use all two perspective cameras along with side fisheye cameras. For evaluation, we rely on provided ground-truth instance masks (KITTI-360)/projected 3D bounding boxes (Pandaset) with motion labels to identify true dynamic objects. We refer to the supplementary for additional details on the selection of the training sequences and data pre-processing.

Table 1: Quantitative comparisons with SoTA methods on KITTI-360 and Pandaset.

		Image Reconstruction			Static NVS			Motion segmentation		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	Recall↑	IoU↑	F1↑
KITTI-360	RobustNeRF [34]†	-	-	-	19.50	0.722	0.282	-	-	-
	D^{2}NeRF [47]	23.86	0.688	0.232	18.70	0.588	0.309	12.04	4.67	7.41
	SUDS [43]	**28.16**	<u>0.836</u>	<u>0.175</u>	18.81	0.687	0.298	**93.89**	5.35	9.34
	EmerNeRF [52]	25.49	0.798	0.181	<u>20.59</u>	<u>0.733</u>	<u>0.249</u>	57.74	<u>12.56</u>	<u>19.92</u>
	RoDUS (ours)	<u>26.47</u>	**0.857**	**0.148**	**23.43**	**0.802**	**0.181**	<u>79.45</u>	**64.84**	**76.52**
Pandaset	RobustNeRF [34]†	-	-	-	23.41	0.764	0.197	-	-	-
	D^{2}NeRF [47]	25.25	0.692	0.291	23.38	0.682	0.292	-	-	-
	SUDS [43]	**30.28**	<u>0.851</u>	<u>0.138</u>	25.34	0.788	0.182	-	-	-
	EmerNeRF [52]	<u>29.51</u>	0.841	0.155	**27.47**	<u>0.811</u>	<u>0.166</u>	-	-	-
	RoDUS (ours)	28.39	**0.877**	**0.135**	<u>25.83</u>	**0.825**	**0.156**	-	-	-

† As RobustNeRF rejects distractors during training, full image reconstruction metrics are not included.

Baseline. We benchmark RoDUS against 4 different state-of-the-art methods for static scene extraction (RobustNeRF [34]) and dynamic scene rendering (D^{2}NeRF [47], SUDS [43], and EmerNeRF [52]). Evaluations focus on two key factors: the quality of novel view synthesis of the static image and the segmentation masks of dynamic objects. While our primary focus is on static-dynamic disentanglement, RoDUS's design allows it to represent dynamic scenes holistically so the quality of the overall image is also considered. We showcase the performance of our method on three representative scenarios in Fig. 5

4.2   Static-Dynamic Decomposition

We report PSNR, SSIM [46], and LPIPS [57] for both image reconstruction and static NVS tasks. We create a validation set by splitting every 8th image (sorted by timestamps) per camera. In the image reconstruction task, we utilize all available samples of the training set for evaluation. In the static NVS task, we compare the image rendered from the static branch with the ground truth with all true dynamic objects masked out. As illustrated in Fig. 6 and Tab. 1, other methods achieve good rendering quality in the composed image. However, only EmerNeRF and RoDUS achieve reasonable results when rendering each component separately. We also lead in SSIM and LPIPS.

4.3   Dynamic Objects Segmentation

For this task, we obtain the motion mask M^D by rendering dynamic opacities and applying a threshold $\mathcal{T}_{O^D} = 0.5$. We report mask Recall, IoU, and F1-score on training views. We only evaluate on KITTI-360 due to the absence of ground-truth instance masks in Pandaset. In Fig. 7, we explore various scenarios to understand how the ego-actor moves in relation to other moving objects and how it observes the scenes that could impact the decomposition. Particularly,
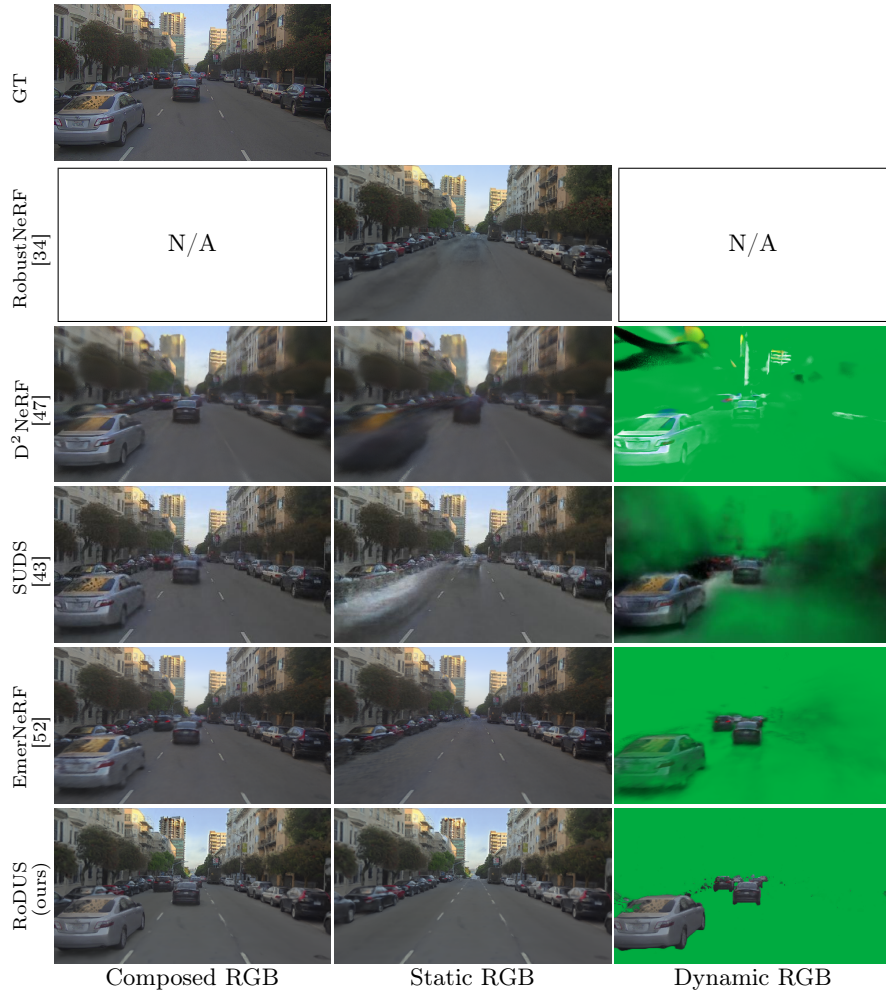
Fig. 6: Reconstruction and decomposition qualitative results comparisons on Pandaset. D^{2}NeRF struggles in intricate driving contexts and produces poor results. SUDS faces difficulties in distinguishing between ego-motion and object-motion due to its reliance on optical flow supervision, leading to a large number of cloudy artifacts in the dynamic image to match the ground truth optical flows. RobustNeRF successfully reconstructs the background with no dynamic objects remaining, but it suffers from saturation, leading to degraded results (as explained in Sec. 3.3). We can see that the separation quality is still suboptimal. In contrast, RoDUS successfully decouples dynamic objects and achieves a clean background with no artifacts remaining.
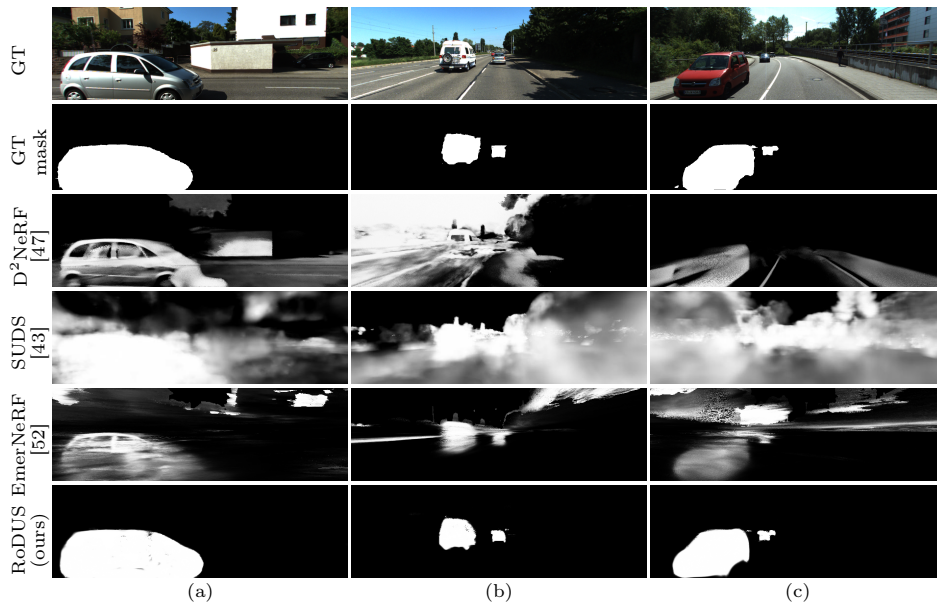
Fig. 7: Motion segmentation qualitative results comparison on KITTI-360 before thresholding. We present results in three scenarios: (a) The ego-car stops in front of an intersection; (b) Vehicles move in the same direction as the ego-car; and (c) Vehicles move in the opposite direction toward the ego-car.

case (a) is likely the easiest as the background is broadly observed for a long period, we can still notice a good separation in D^2NeRF, SUDS's results. In cases (b) and (c), when the ego actor is in motion, most methods suffer from incorrect segmentation, while our semantic guidance forces the model to learn the correct dynamic objects. SUDS tends to produce large dynamic areas, resulting in very high Recall but low IoU and F1, yet EmerNeRF exhibits sensitivity to lighting variations. Correspondingly, Tab. 1 indicates that RoDUS's metrics outperform the others in both tasks.

4.4   Ablation Study

We analyze our design choices in Tab. 2. The use of road, sky regularization, and robust initialization terms support the geometry, especially in regions occluded by dynamic objects, thus improving the overall quality. Please refer to Fig. 4 and Fig. 8 for visual comparisons.

4.5   Limitation

In general, RoDUS works best with scenes where labels belonging to potentially movable objects are well-defined. Nonetheless, if the semantic ground truth is absent or heavily mislabeled, it moderately impacts performance. Moreover, our

Table 2: Ablation study. Impact of our design choice on static NVS task.

Ablation	PSNR↑	SSIM↑	LPIPS↓
Full model	25.94	0.814	0.170
w/o $\mathcal{L}_d$	-0.80	-0.032	+0.023
w/o $\mathcal{L}_{sem}$	-1.67	-0.080	+0.078
w/o $\mathcal{L}_{robust}$	-0.91	-0.047	+0.067
w/o $\mathcal{L}_{road}$	-0.84	-0.035	+0.026
w/o $\mathcal{L}_{\sigma^D}$	-0.74	-0.030	+0.024
w/o FG mask	-0.67	-0.035	+0.023

w/o FG mask   w/o $\mathcal{L}_{sem}$

w/o $\mathcal{L}_{robust}$ Full model
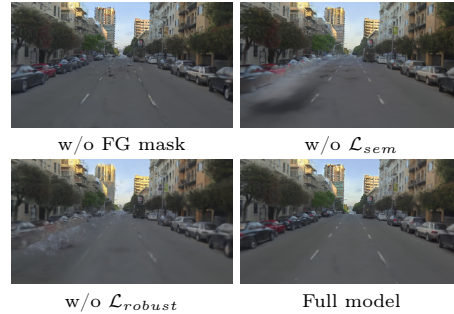
Fig. 8: Components ablation on $\hat{I}^S$.

approach can't handle areas that lack observations or have been highly occluded most of the time within the sequence. This is particularly notable by large or slow-moving objects in Fig. 9. Exploring a model that is capable of uncertainty modeling to incorrect or anomalous pixels and inpainting these missing regions based on prior knowledge will be interesting as a future research direction.

GT $\hat{I}^S$ $\hat{S}^S$

$\hat{I}$ $\hat{I}^D$ M^D
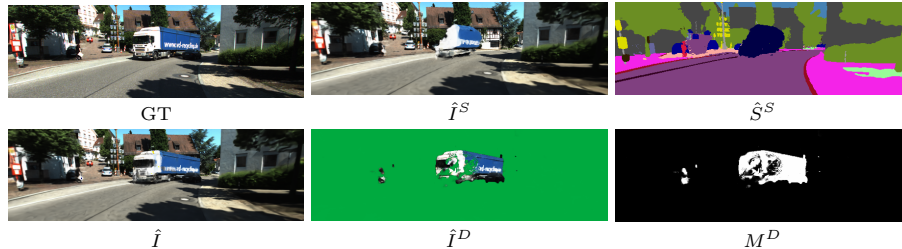
Fig. 9: Failure case. The static field is unable to recover regions that are highly occluded by the truck due to its lack of generalizing ability to unseen backgrounds. On the other hand, the dynamic field doesn't support object-level connection, leading to only half of the truck being visible in dynamic outputs.

5   Conclusion

In this paper, we presented RoDUS, a method for disentangling dynamic scenes using neural representation. Our approach effectively separated moving components from the scene without the need for manual labeling. To enhance disentangling efficiency, we proposed a robust training strategy and incorporated semantic reasoning, resulting in high-quality background view synthesis and accurate segmentation. Extensive experiments on challenging autonomous driving datasets showcased that RoDUS surpasses current state-of-the-art NeRF frameworks.

References

1. Barron, J.T.: A general and adaptive robust loss function. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4331–4339 (2019)
2. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5470–5479 (2022)
3. Cao, Z., Kar, A., Hane, C., Malik, J.: Learning independent object motion from unlabelled stereoscopic videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5594–5603 (2019)
4. Chen, X., Zhang, Q., Li, X., Chen, Y., Feng, Y., Wang, X., Wang, J.: Hallucinated neural radiance fields in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12943–12952 (2022)
5. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1290–1299 (2022)
6. Chetverikov, D., Svirko, D., Stepanov, D., Krsek, P.: The trimmed iterative closest point algorithm. In: 2002 International Conference on Pattern Recognition. vol. 3, pp. 545–548. IEEE (2002)
7. Deka, M.S., Sang, L., Cremers, D.: Erasing the ephemeral: Joint camera refinement and transient object removal for street view synthesis. arXiv preprint arXiv:2311.17634 (2023)
8. Deng, K., Liu, A., Zhu, J.Y., Ramanan, D.: Depth-supervised nerf: Fewer views and faster training for free. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12882–12891 (2022)
9. Fu, X., Zhang, S., Chen, T., Lu, Y., Zhu, L., Zhou, X., Geiger, A., Liao, Y.: Panoptic nerf: 3d-to-2d label transfer for panoptic urban scene segmentation. In: 2022 International Conference on 3D Vision (3DV). pp. 1–11. IEEE (2022)
10. Guo, J., Deng, N., Li, X., Bai, Y., Shi, B., Wang, C., Ding, C., Wang, D., Li, Y.: Streetsurf: Extending multi-view implicit surface reconstruction to street views. arXiv preprint arXiv:2306.04988 (2023)
11. Herau, Q., Piasco, N., Bennehar, M., Roldão, L., Tsishkou, D., Migniot, C., Vasseur, P., Demonceaux, C.: Moisst: Multi-modal optimization of implicit scene for spatiotemporal calibration. In: International Conference on Intelligent Robots and Systems (IROS) (2023)
12. Herau, Q., Piasco, N., Bennehar, M., Roldão, L., Tsishkou, D., Migniot, C., Vasseur, P., Demonceaux, C.: Soac: Spatio-temporal overlap-aware multi-sensor calibration using neural radiance fields. arXiv preprint arXiv:2311.15803 (2023)
13. Irshad, M.Z., Zakharov, S., Liu, K., Guizilini, V., Kollar, T., Gaidon, A., Kira, Z., Ambrus, R.: Neo 360: Neural fields for sparse view synthesis of outdoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9187–9198 (2023)
14. Kobayashi, S., Matsumoto, E., Sitzmann, V.: Decomposing nerf for editing via feature field distillation. Advances in Neural Information Processing Systems **35**, 23311–23330 (2022)
15. Kundu, A., Genova, K., Yin, X., Fathi, A., Pantofaru, C., Guibas, L.J., Tagliasacchi, A., Dellaert, F., Funkhouser, T.: Panoptic neural fields: A semantic object-

aware neural scene representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12871–12881 (2022)

16. Li, H., Gordon, A., Zhao, H., Casser, V., Angelova, A.: Unsupervised monocular depth learning in dynamic scenes. In: Conference on Robot Learning. pp. 1908–1917. PMLR (2021)

17. Li, T., Slavcheva, M., Zollhoefer, M., Green, S., Lassner, C., Kim, C., Schmidt, T., Lovegrove, S., Goesele, M., Newcombe, R., et al.: Neural 3d video synthesis from multi-view video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5521–5531 (2022)

18. Li, Z., Niklaus, S., Snavely, N., Wang, O.: Neural scene flow fields for space-time view synthesis of dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6498–6508 (2021)

19. Liao, Y., Xie, J., Geiger, A.: Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(3), 3292–3310 (2022)

20. Liu, Y.L., Gao, C., Meuleman, A., Tseng, H.Y., Saraf, A., Kim, C., Chuang, Y.Y., Kopf, J., Huang, J.B.: Robust dynamic radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13–23 (2023)

21. Martin-Brualla, R., Radwan, N., Sajjadi, M.S., Barron, J.T., Dosovitskiy, A., Duckworth, D.: Nerf in the wild: Neural radiance fields for unconstrained photo collections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7210–7219 (2021)

22. Mattheus, J., Grobler, H., Abu-Mahfouz, A.M.: A review of motion segmentation: Approaches and major challenges. In: 2020 2nd International multidisciplinary information technology and engineering conference (IMITEC). pp. 1–8. IEEE (2020)

23. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)

24. Müller, T.: tiny-cuda-nn (2021), `https://github.com/NVlabs/tiny-cuda-nn`

25. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Transactions on Graphics (ToG) **41**(4), 1–15 (2022)

26. Neuhold, G., Ollmann, T., Rota Bulo, S., Kontschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: Proceedings of the IEEE international conference on computer vision. pp. 4990–4999 (2017)

27. Ost, J., Mannan, F., Thuerey, N., Knodt, J., Heide, F.: Neural scene graphs for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2856–2865 (2021)

28. Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., Martin-Brualla, R.: Nerfies: Deformable neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5865–5874 (2021)

29. Park, K., Sinha, U., Hedman, P., Barron, J.T., Bouaziz, S., Goldman, D.B., Martin-Brualla, R., Seitz, S.M.: Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. arXiv preprint arXiv:2106.13228 (2021)

30. Park, S., Son, M., Jang, S., Ahn, Y.C., Kim, J.Y., Kang, N.: Temporal interpolation is all you need for dynamic neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4212–4221 (2023)

31. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-

performance deep learning library. Advances in neural information processing systems **32** (2019)

32. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: D-nerf: Neural radiance fields for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10318–10327 (2021)

33. Rematas, K., Liu, A., Srinivasan, P.P., Barron, J.T., Tagliasacchi, A., Funkhouser, T., Ferrari, V.: Urban radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12932–12942 (2022)

34. Sabour, S., Vora, S., Duckworth, D., Krasin, I., Fleet, D.J., Tagliasacchi, A.: Robustnerf: Ignoring distractors with robust losses. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20626–20636 (2023)

35. Saunders, K., Vogiatzis, G., Manso, L.J.: Dyna-dm: Dynamic object-aware self-supervised monocular depth maps. In: 2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC). pp. 10–16. IEEE (2023)

36. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016)

37. Schönberger, J.L., Zheng, E., Frahm, J.M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14. pp. 501–518. Springer (2016)

38. Sharma, P., Tewari, A., Du, Y., Zakharov, S., Ambrus, R.A., Gaidon, A., Freeman, W.T., Durand, F., Tenenbaum, J.B., Sitzmann, V.: Neural groundplans: Persistent neural scene representations from a single image. In: International Conference on Learning Representations (2023)

39. Siddiqui, Y., Porzi, L., Bulò, S.R., Müller, N., Nießner, M., Dai, A., Kontschieder, P.: Panoptic lifting for 3d scene understanding with neural fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9043–9052 (2023)

40. Tonderski, A., Lindström, C., Hess, G., Ljungbergh, W., Svensson, L., Petersson, C.: Neurad: Neural rendering for autonomous driving. arXiv preprint arXiv:2311.15260 (2023)

41. Tschernezki, V., Laina, I., Larlus, D., Vedaldi, A.: Neural feature fusion fields: 3d distillation of self-supervised 2d image representations. In: 2022 International Conference on 3D Vision (3DV). pp. 443–453. IEEE (2022)

42. Tschernezki, V., Larlus, D., Vedaldi, A.: Neuraldiff: Segmenting 3d objects that move in egocentric videos. In: 2021 International Conference on 3D Vision (3DV). pp. 910–919. IEEE (2021)

43. Turki, H., Zhang, J.Y., Ferroni, F., Ramanan, D.: Suds: Scalable urban dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12375–12385 (2023)

44. Wang, F., Chen, Z., Wang, G., Song, Y., Liu, H.: Masked space-time hash encoding for efficient dynamic scene reconstruction. arXiv preprint arXiv:2310.17527 (2023)

45. Wang, F., Louys, A., Piasco, N., Bennehar, M., Roldão, L., Tsishkou, D.: Planerf: Svd unsupervised 3d plane regularization for nerf large-scale scene reconstruction. arXiv preprint arXiv:2305.16914 (2023)

46. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)

47. Wu, T., Zhong, F., Tagliasacchi, A., Cole, F., Oztireli, C.: Dˆ 2nerf: Self-supervised decoupling of dynamic and static objects from a monocular video. Advances in Neural Information Processing Systems **35**, 32653–32666 (2022)
48. Wu, Z., Liu, T., Luo, L., Zhong, Z., Chen, J., Xiao, H., Hou, C., Lou, H., Chen, Y., Yang, R., et al.: Mars: An instance-aware, modular and realistic simulator for autonomous driving. In: CAAI International Conference on Artificial Intelligence. pp. 3–15. Springer Nature Singapore Singapore (2023)
49. Xiao, P., Shao, Z., Hao, S., Zhang, Z., Chai, X., Jiao, J., Li, Z., Wu, J., Sun, K., Jiang, K., et al.: Pandaset: Advanced sensor suite dataset for autonomous driving. In: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC). pp. 3095–3101. IEEE (2021)
50. Xu, Y., Chai, M., Shi, Z., Peng, S., Skorokhodov, I., Siarohin, A., Yang, C., Shen, Y., Lee, H.Y., Zhou, B., et al.: Discoscene: Spatially disentangled generative radiance fields for controllable 3d-aware scene synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4402–4412 (2023)
51. Yan, Y., Lin, H., Zhou, C., Wang, W., Sun, H., Zhan, K., Lang, X., Zhou, X., Peng, S.: Street gaussians for modeling dynamic urban scenes. arXiv preprint arXiv:2401.01339 (2024)
52. Yang, J., Ivanovic, B., Litany, O., Weng, X., Kim, S.W., Li, B., Che, T., Xu, D., Fidler, S., Pavone, M., et al.: Emernerf: Emergent spatial-temporal scene decomposition via self-supervision. arXiv preprint arXiv:2311.02077 (2023)
53. Yang, Y., Yang, Y., Guo, H., Xiong, R., Wang, Y., Liao, Y.: Urbangiraffe: Representing urban scenes as compositional generative neural feature fields. arXiv preprint arXiv:2303.14167 (2023)
54. Yang, Z., Chen, Y., Wang, J., Manivasagam, S., Ma, W.C., Yang, A.J., Urtasun, R.: Unisim: A neural closed-loop sensor simulator. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1389–1399 (2023)
55. Ye, W., Yu, X., Lan, X., Ming, Y., Li, J., Bao, H., Cui, Z., Zhang, G.: Deflowslam: Self-supervised scene motion decomposition for dynamic dense slam. arXiv preprint arXiv:2207.08794 (2022)
56. Yuan, W., Lv, Z., Schmidt, T., Lovegrove, S.: Star: Self-supervised tracking and reconstruction of rigid objects in motion with neural rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13144–13152 (2021)
57. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)
58. Zhang, X., Kundu, A., Funkhouser, T., Guibas, L., Su, H., Genova, K.: Nerflets: Local radiance fields for efficient structure-aware 3d scene representation from 2d supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8274–8284 (2023)
59. Zhi, S., Laidlow, T., Leutenegger, S., Davison, A.J.: In-place scene labelling and understanding with implicit scene representation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15838–15847 (2021)
60. Zhou, X., Lin, Z., Shan, X., Wang, Y., Sun, D., Yang, M.H.: Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. arXiv preprint arXiv:2312.07920 (2023)