Neural Fields for Co-Reconstructing 3D Objects from Incidental 2D Data

Dylan Campbell* Eldar Insafutdinov* João F. Henriques Andrea Vedaldi
Australian National University
University of Oxford
dylan.campbell@anu.edu.au
{eldar,joao,vedaldi}@robots.ox.ac.uk

Abstract

We ask whether 3D objects can be reconstructed from real world data collected for some other purpose, such as autonomous driving or augmented reality, thus inferring objects only incidentally. *3D reconstruction from incidental data is a major challenge because, in addition to significant noise, only a few views of each object are observed, which are insufficient for reconstruction. We approach this problem as a co-reconstruction task, where multiple objects are reconstructed together, learning shape and appearance priors for regularization. In order to do so, we introduce a neural radiance field that is conditioned via an attention mechanism on the identity of the individual objects. We further disentangle shape from appearance and diffuse color from specular color via an asymmetric two-stream network, which factors shared information from instance-specific details. We demonstrate the ability of this method to reconstruct full 3D objects from partial, incidental observations in autonomous driving and other datasets.*

1. Introduction

The development of technologies such as autonomous driving and augmented reality means that there is an enormous quantity of videos that capture the real world [4, 11, 17]. While this data is collected for a specific purpose, such as controlling a car or augmenting an environment, it also *incidentally* contains a lot of information about the world and the 3D objects therein. In this paper, we ask if it is possible to learn models of 3D objects from data collected for some other purpose, inferring them only incidentally.

In practice, this task is quite challenging. Even modern 3D reconstruction algorithms [30, 41] generally assume that the input images *focus* on the object of interest, and usually 'circumnavigate' it, providing 360° coverage. Incidental recordings only catch glimpses of the objects from a small range of viewpoints, usually a single side, and are also affected by occlusion and noise (Fig. 1).

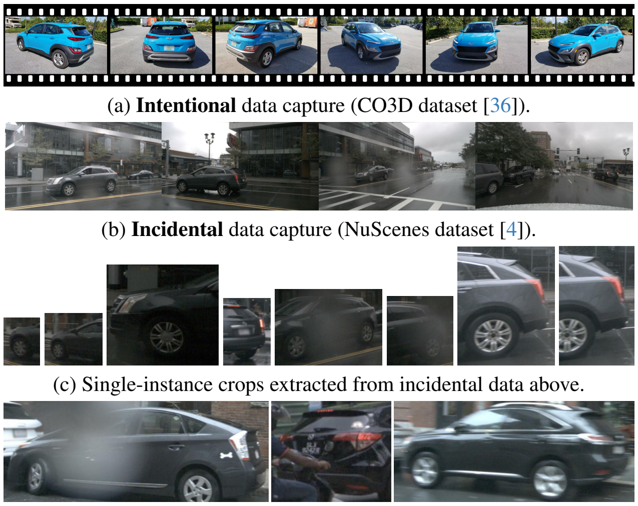

(a) **Intentional** data capture (CO3D dataset [36]).
(b) **Incidental** data capture (NuScenes dataset [4]).
(c) Single-instance crops extracted from incidental data above.
(d) Challenges: lens droplets, dynamic occlusions, motion blur.

Figure 1. Comparing *intentional* and *incidental* data. (a) Data captured intentionally for 3D reconstruction, like the CO3D dataset [36], circumnavigates and focuses on the object, is carefully collected in good conditions, and has accurate poses and masks, but is scarce. (b–c) Data captured incidentally, like the NuScenes dataset [4] where vehicle-mounted cameras catch glancing views of objects, is plentiful, but comes with many challenges. (d) These include incomplete capture (viewing a single side), low resolutions, motion blur, static/dynamic occlusions, and adverse conditions. Our model produces complete reconstructions from incidental and incomplete data, by co-reconstructing many objects simultaneously and learning helpful shape and appearance priors.

Consider as a representative of this task inferring 3D objects from data obtained for autonomous driving, like the NuScenes dataset [4], with video collected by vehicle-mounted cameras. This data contains abundant information about 3D objects such as vehicles. However, extracting a model of such objects faces several challenges, including highly reflective and low-textured surfaces (*e.g.*, mirrors, windows, and metallic paint), static and dynamic occlusions, other dynamic effects (*e.g.*, motion blur, rotating wheels, and moving reflections), adverse weather conditions, sensor perturbations (*e.g.*, water droplets and grime),

[*] Both authors contributed equally to this research.

and incorrectly estimated camera poses and instance masks. Any system that seeks to take advantage of such data must also be robust to these nuisance factors.

The partiality of the observations means that the images available for any given object are insuffcent for reconstructing it. To solve this reconstruction-at-a-glance problem, we propose a co-reconstruction setting which learns category-specific shape and appearance priors, such as symmetries, material properties, and part colors. However, learning these intrinsic priors is non-trivial, since the space of possible combinations of shape and texture is combinatorial and there are many confounding extrinsic factors, including lighting, shadows, reflections, and dynamic effects.

Our model overcomes these problems by disentangling shape from appearance and diffuse color from specular color, allowing the common properties to be shared across instances. This enables our model to accurately co-reconstruct a set of instances despite only having seen glimpses of each, as shown in Fig. 1, overcoming a major shortcoming of neural reconstruction approaches like NeRF [30] that can only reconstruct surfaces seen in the training images. We also propose a new parameterization of the space of co-reconstructed NeRFs that leverages attention conditioning to outperform off-the-shelf solutions.

Hence, our **contributions** are: (1) a novel problem setting, co-reconstructing 3D objects captured incidentally in the real world; (2) a disentangled neural fields decoder model with an asymmetric two-stream design; and (3) a hierarchical, efficiently-conditioned, and decorrelated decoder backbone. We demonstrate complete and high-fidelity reconstructions, allowing for convincing novel view synthesis of unseen sides of objects, and achieve state-of-the-art co-reconstruction results.

2. Related Work

We review the neural rendering work closest to ours; a general overview can be obtained from surveys [44, 46].

Neural rendering and reconstruction. The Neural Radiance Fields (NeRF) method of [30] and related work [1, 27, 29, 41, 49, 52, 53] represent the color and opacity of each point in space using a neural network for each scene, and render images using the emission-absorption model. Our model also uses neural rendering, but we extend the setup to category-level co-reconstruction in order to learn shape and appearance priors in common to many objects.

To address the difficulties associated with modeling complex and reflective surfaces, a number of works explore more sophisticated lighting models [2, 3, 9, 43, 48, 55]. For example, Ref-NeRF [48] optimizes a NeRF-like model with a reflected radiance parameterization that better captures light transport. Our model also disentangles the diffuse and specular components of appearance via an additive color model. However, our motivation is to allow the network to share information about material properties between instances, for which specular effects are nuisance variables.

Another line of work investigates optimization-efficient representations for learning neural fields. The memory–compute trade-off can be altered by replacing an MLP-parameterized neural field with voxels [13], hash tables [33], triplanes [5, 42], or tensor decompositions [8]. For example, TensoRF [8] decomposes the neural field grid into vector–matrix pairs, which can be optimized rapidly. We use the similar triplane representation [5], which consists of three axis-aligned orthogonal feature planes. Unlike TensoRF, we use a triplane decoder, rather than optimizing the triplane parameters directly, allowing us to learn priors.

Mesh reconstruction. Category-level reconstruction has been previously explored for meshes [15, 22, 26, 50, 51]. For example, Ye *et al*. [51] infer shape, texture, and pose from a single image via an auto-encoding network with photometric and adversarial losses. However, the approach cannot take advantage of additional test views and has lower tolerance for the occlusion, blur, and noisy masks present in incidental data, preferring synthetic and curated collections. These approaches tend to excel in the sparse data regime where a strong shape prior is helpful, but are less detailed and photorealistic than neural field approaches.

Conditional neural fields. Multiple instances of the same category can be handled by conditioning the reconstruction network on per-instance latent codes. Several approaches [6, 34, 39] learn generative adversarial networks [16] with a neural field generator conditioned on instance-specific codes via concatenation [34, 39, 45] or normalization [6]. For example, π-GAN [6] uses Adaptive Instance Normalization (AdaIN) [12, 20] to condition SIREN-based [40] implicit radiance fields. EG3D [5], EpiGRAF [42], and GET3D [14] instead generate a triplane using a StyleGAN2 [23] synthesis network, conditioned via AdaIN. Like these works, we use a triplane generator, but propose cross-attention for efficient triplane conditioning.

Non-adversarial approaches include those that condition on local image features [36, 52], and those that condition on an instance via decoding or auto-encoding [21, 31, 32]. CodeNeRF [21] conditions intermediate features of a NeRF-like MLP decoder on mapped latent codes through residual connections. In contrast, AutoRF [32] and UNICORN [31] use auto-encoders to learn a low-dimensional intermediate latent feature from a single image, which can be decoded into a 3D reconstruction and rendered from any viewpoint. Our conditional decoding approach is most similar to CodeNeRF, but substantially improves the quality of the reconstructions due to its disentangled triplane decoder design and efficient cross-attention conditioning.

3. Neural Fields for Co-Reconstruction

We cast our problem as image-based reconstruction and approach it using a neural rendering formulation: given a certain number of views of a target 3D object, we fit a density and radiance field represented using a neural network, thus reconstructing the object. Where we depart from standard neural rendering solutions (Sec. 3.1) is how we address the lack of a sufficient number of views of the object: we do so by *co-reconstructing* many different objects together, so that the reconstructions are better than what would be obtainable from each instance individually.

In order to maximize parameter sharing between different objects, we propose a new *neural field architecture* (Sec. 3.2) and a *disentangled representation* of appearance, material and illumination to factor out instance-specific effects (Sec. 3.3). We also develop a robust learning objective to address other types of noise in the data (Sec. 3.4). A flowchart of our model is shown in Fig. 3.

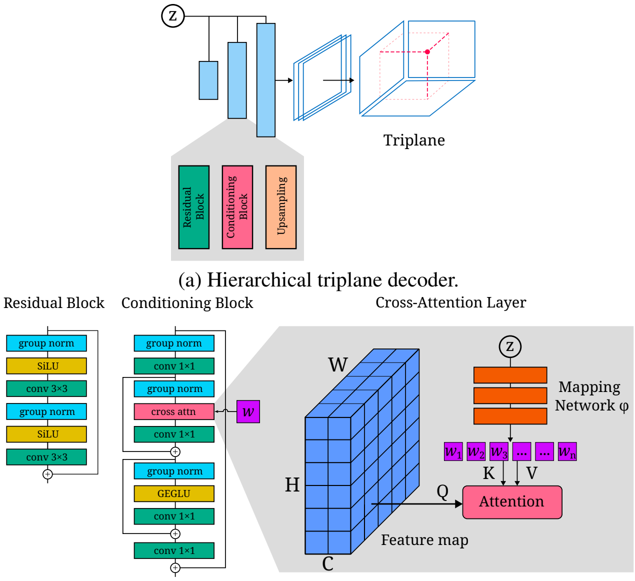
At training time, we optimize the model parameters (shared across instances), the extrinsic camera parameters, and the shape, appearance, and directional codes (shared within instances) to best reconstruct the data. Furthermore, a small "view code" is also allowed to vary between frames, to account for minor appearance changes across time (*e.g.*, auto-exposure, wheel rotation, and motion blur).

Once learned, we can utilize our model in two ways. First, we can generate novel views of the objects in the training set. To do so, we optimize the extrinsic camera parameters and the low-dimensional view code with respect to a held-out set of test images and evaluate the quality of the rendered images. Second, we can reconstruct a new instance of a previously-trained category. To do so we optimize the camera parameters and codes with respect to the set of training images, keeping the model parameters fixed, and evaluate performance as above.

Formally, given a set $\{x_i\}_{i=1}^{M}$ of object instances of a given category, where each instance x_i consists of a set $\{(\boldsymbol{I}_{ij}, \boldsymbol{M}_{ij}, \boldsymbol{T}_{ij}, \boldsymbol{K}_{ij})\}_{j=1}^{N_i}$ of N_i multi-view images $\boldsymbol{I}$, approximate binary object masks $\boldsymbol{M}$, approximate camera extrinsic (pose) matrices $\boldsymbol{T}$ relative to the object, and intrinsic matrices $\boldsymbol{K}$, the task is to co-reconstruct the shape and view-dependent appearance of every object. Estimated camera-to-object pose parameters can be obtained by detecting and tracking 3D bounding boxes of the (potentially moving) object of interest, aided by any available odometry.

3.1. Background: Neural rendering

In neural rendering [30], a function $\phi : \mathbb{R}^3 \times \mathbb{S}^3 \to \mathbb{R}^+ \times \mathbb{R}^3$, parameterized by the weights of a neural network, maps a 3D point $\boldsymbol{x} \in \mathbb{R}^3$ and view direction $\boldsymbol{d}$ to a volume density σ and color $\boldsymbol{c} \in [0,1]^3$. To learn this function from posed images, colors are accumulated along a pixel ray emanating from a camera centre $\boldsymbol{o}$ in the direction $\boldsymbol{d}$. For K point samples $\{\boldsymbol{x}_i = \boldsymbol{o} + t_i\boldsymbol{d} \mid i = 1, \ldots, K;\ t_i < t_{i+1}\}$, with gaps between adjacent samples $\delta_i = t_{i+1} - t_i$, the rendered color is given by



(a) Hierarchical triplane decoder.

(b) Decoder block structure.

Figure 2. Attention-conditioned triplane decoder. The network structure is hierarchical, with coarse-to-fine decoding and conditioning so that coarse shape and appearance is learned before the details. It uses cross-attention conditioning to make most efficient use of the latent parameters, and uses group convolutions to avoid introducing spurious correlations between triplane elements.

$$\hat{\boldsymbol{c}}(\boldsymbol{o}, \boldsymbol{d}) = \sum_{i=1}^{K} \exp\left(-\sum_{j=1}^{i-1} \sigma_j \delta_j\right) \left(1 - \exp(-\sigma_i \delta_i)\right) \boldsymbol{c}_i. \quad (1)$$

The parameters of the network ϕ are then learned by minimizing the distance between the training input images and the images generated from Eq. (1).

3.2. Attention-conditioned neural fields

Our first contribution is a new neural radiance field architecture (Figs. 2a and 2b) that can efficiently model multiple objects by modulating shared parameters to account for instance- and view-specific differences between objects.

We learn a mapping from a latent vector $\boldsymbol{z}$, coding for a specific 3D object, into a triplane representation [5] of the neural radiance field. We choose a triplane for its efficiency and because it can be processed using 2D convolutions. The backbone of the mapping, illustrated in Fig. 2a, is a coarse-to-fine U-Net [38] augmented with conditioning layers, reminiscent of Stable Diffusion [37]. A key design decision was to use *cross-attention for conditioning* [47], as shown in Fig. 2b, since we empirically found it better at fitting large and complex datasets [35]. Specifically, we

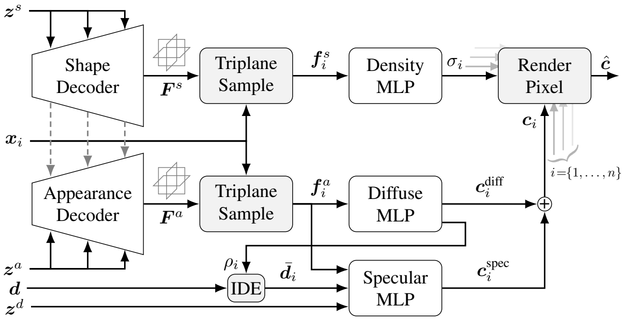


Figure 3. The Disentangled Neural Field Decoder (DNFD) model. First, shape and appearance triplanes ($\boldsymbol{F}^s$ and $\boldsymbol{F}^a$) are generated from two decoders, one conditioned on a per-instance latent shape code $\boldsymbol{z}^s$ and the other on a per-instance appearance code $\boldsymbol{z}^a$. The appearance decoder is conditioned on the shape decoder via uni-directional residual connections. The triplanes are sampled at a 3D point $\boldsymbol{x}_i$ to obtain shape and appearance features ($\boldsymbol{f}_i^s$ and $\boldsymbol{f}_i^a$). Small MLPs compute the density σ_i from $\boldsymbol{f}_i^s$ and the diffuse color $\boldsymbol{c}_i^{\text{diff}}$ from $\boldsymbol{f}_i^a$. Next, the appearance feature $\boldsymbol{f}_i^a$, encoded direction $\bar{\boldsymbol{d}}_i$, and directional latent code $\boldsymbol{z}^d$ are passed to a small MLP to predict the specular color $\boldsymbol{c}_i^{\text{spec}}$. The view direction $\boldsymbol{d}$ is encoded with Integrated Directional Encoding (IDE) [48], which also takes the predicted scalar roughness ρ_i as input. The directional code helps factor out nuisance variables like reflections, autoexposure changes, and dynamic effects that are not shareable between instances or frames. Finally, the diffuse and specular colors are summed and integrated along the ray to render the pixel color $\hat{\boldsymbol{c}}$.

project the code $\boldsymbol{z}$ to a sequence of tokens $\boldsymbol{w}_i$ via a mapping network, and use them as cross-attention keys and values.

In order to process the triplanes with a single application of a 2D convnet, we stack them along the channel dimension. However, since they do not align spatially because they code different coordinate planes, we process them using group convolutions [25], which reduces the risk of learning spurious correlations between them [5, 14, 42]. This removes a harmful inductive bias and greatly improves parameter efficiency.

3.3. Separating instance and view-specific factors

In order to maximize information sharing between different object instances and different views of the same object, the model must be able to factor information that can and cannot be shared. Disentangling shape and appearance allows the network to reason about shape independently from appearance, facilitating the sharing of information between instances. For example, the parameters used to model the shape of a Mini Cooper should not depend on its paint job or the lighting conditions. However, the appearance at any 3D coordinate does depend on the shape. For example, the color depends on where the coordinate lies on the object's surface, on the normal directions, and on shadows cast by non-convex geometries.

Two-stream decoder. To achieve this disentanglement, we propose a novel asymmetric two-stream decoder design. As shown in Fig. 3, our model has a shape decoder that upsamples a constant feature map, conditioned on a per-instance shape code $\boldsymbol{z}^s$, to produce a triplane of three orthogonal shape feature maps [5]. For each sample point in 3D, a shape feature is obtained from this triplane using bilinear interpolation and is decoded to the volume density σ_i using a small fully-connected network. A second decoder for appearance is conditioned on a per-instance appearance code $\boldsymbol{z}^a$, generating an appearance triplane. Crucially, the appearance decoder receives information from the shape decoder via uni-directional residual connections, so that the shape informs appearance, but not vice versa.

View-dependent reflected colors. View-dependent effects, such as reflections and specular highlights, as well as frame-dependent effects, such as exposure and dynamic color changes, are not shareable between instances and can be considered nuisance variables for this task. To address this, we factorize the color into diffuse (view-independent) and specular (view-dependent) components. The diffuse component $\boldsymbol{c}_i^{\text{diff}}$ is defined as the average color of a 3D point across all observed viewing directions, under the lighting conditions for that instance. The specular component $\boldsymbol{c}_i^{\text{spec}}$ captures view- and frame-dependent effects by conditioning on a directional latent code $\boldsymbol{z}^d$ with a small number of parameters that vary per-frame.

Specifically, we bilinearly interpolate the appearance triplane $\boldsymbol{F}^a$ at the 3D position $\boldsymbol{x}_i$ to obtain an appearance feature $\boldsymbol{f}_i^a$, encoding the local texture, normal, lighting, and reflectivity (Fig. 3, bottom left). A small three-layer MLP decodes this into the diffuse color $\boldsymbol{c}_i^{\text{diff}}$ and the roughness ρ_i. The latter is used to encode the direction vector $\boldsymbol{d}$ via the spherical harmonics-based Integrated Directional Encoding (IDE) [48]. Finally, another three-layer MLP, conditioned on the directional latent code $\boldsymbol{z}^d$, decodes the appearance feature $\boldsymbol{f}_i^a$ and the encoded direction vector $\bar{\boldsymbol{d}}_i$ into the specular color $\boldsymbol{c}_i^{\text{spec}}$. Our additive color formation model is

$$\boldsymbol{c}_i = \boldsymbol{c}_i^{\text{diff}}(\boldsymbol{f}_i^a(\boldsymbol{x}_i)) + \boldsymbol{c}_i^{\text{spec}}(\boldsymbol{f}_i^a(\boldsymbol{x}_i), \bar{\boldsymbol{d}}_i), \tag{2}$$

where $\boldsymbol{c}_i$ is the estimated color of the 3D point $\boldsymbol{x}_i$, $\boldsymbol{c}_i^{\text{diff}}$ is the direction-invariant diffuse color of the material, depending on the position and shape only, and $\boldsymbol{c}_i^{\text{spec}}$ is the specular color of the reflected light, given an encoded view direction $\bar{\boldsymbol{d}}_i$. To encourage disentanglement, we apply a loss to match the diffuse color $\boldsymbol{c}_i^{\text{diff}}$, rendered along the ray, to the ground-truth color. This encourages the network to predict $\boldsymbol{c}_i^{\text{diff}}$ as the color averaged over all observed viewing directions.

3.4. A robust loss for noisy incidental data

To compensate for low-quality images and poorly-estimated camera-to-object poses and masks, as are common in incidental data, robust loss functions are critical. We use the masked L_1 error between the rendered and ground-truth pixels and the L_1 error between the predicted and (approximate) ground-truth instance mask. No 3D supervision is used, beyond the approximate poses used for initialization. We optimize the network parameters, the per-frame extrinsic camera parameters (rotation and translation), and the per-instance latent codes. The per-pixel color loss is $\mathcal{L}^{\text{color}} = \frac{1}{3}m\|\hat{\boldsymbol{c}} - \boldsymbol{c}\|_1$, where $\hat{\boldsymbol{c}}$ is the predicted color, $\boldsymbol{c}$ is the ground-truth color, and $m \in \{0, 1\}$ is the binary instance mask variable that equals 1 for any pixel of the object instance. The associated mask loss is given by $\mathcal{L}^{\text{mask}} = \|\hat{m} - m\|_1$, where $\hat{m} = 1 - T_K$ is the predicted mask, and T_K is the accumulated transmittance along the ray at the final (K^{th}) sample. We also apply a masked diffuse color loss $\mathcal{L}^{\text{diffuse}}$ with the same form as above. For this loss, the pixel color is rendered without the specular component, encouraging the network to explain as much of the training data as possible without view-dependent effects. Finally, we apply L_2 regularization to the latent codes to discourage overfitting. This is implemented via weight decay with the AdamW optimizer [28]. The total per-pixel loss, with hyperparameters λ, is then

$$\mathcal{L} = \mathcal{L}^{\text{color}} + \lambda^{\text{m}}\mathcal{L}^{\text{mask}} + \lambda^{\text{d}}\mathcal{L}^{\text{diffuse}}. \tag{3}$$

4. Results

4.1. Experimental setup

Datasets. We evaluate our method on the NuScenes dataset [4], released under the CC BY-NC-SA 4.0 License, and the ShapeNet dataset [7], as well as the Woven Planet (Lyft) Level 5 dataset [19] and the ScanNet dataset [11] in the appendix. NuScenes is a large-scale in-the-wild outdoor dataset of 1.4M vehicle-mounted camera images from 1000 driving scenes in Boston and Singapore, with ground-truth camera poses, intrinsics, and 3D bounding box annotations for keyframes. We augment the dataset with approximate instance segmentation masks predicted by Mask2Former [10]. After filtering, we obtain 4157 instances, each with two random test frames withheld.

This real-world dataset was not collected with the intention that it be used for reconstruction, making it particularly challenging for this task. However, we see the size of datasets like this as an opportunity to scale up existing reconstruction models, if the concomitant challenges can be overcome. These include incomplete observations (viewed from one side), significant motion and vibration blur, dynamic and static occlusions, adverse weather, nighttime captures, auto-exposure, widely-varying resolutions, lens droplets and grime, and inconsistent privacy blurring. We apply our method to the vehicle category, which is the best-represented object category in this dataset, and has its own unique challenges: highly reflective and low-textured surfaces (*e.g.*, mirrors, windows, and metallic paint), dynamic parts (*e.g.*, rotating wheels, windscreen wipers, flashing lights), and motion blur.

For the synthetic ShapeNet cars dataset [7], we follow the dataset split (2458 training instances) and rendering protocol of Scene Representation Networks (SRN) [41] but rerender at 4× the resolution with transparency and specularities enabled, bringing the data closer to real conditions and challenges. We also provide the ground-truth depth maps to faciliate geometric evaluation. For the chairs dataset, with its less complex textures, we use the standard SRN dataset [41]. The train and test frames are divided such that they are taken from a strictly different half-space. These experiments therefore assess the ability of a model to *extrapolate* to significantly different viewpoints—visualizing a side of the object it has not seen.

Metrics. We report four metrics to measure the visual and geometric quality of the reconstructions: the perceptual LPIPS distance [54] and the peak signal-to-noise ratio (PSNR) between the masked predicted and ground-truth novel-view images; the intersection-over-union (IoU) between the predicted and ground-truth object masks; and the Fréchet Inception Distance (FID) [18]. PSNR has significant shortcomings as a metric in this setting, because blurring causes a small PSNR change but a large perceptual change. We consider LPIPS to be the more useful measure, alongside the FID for gauging realism, and recommend viewing the video results.

Baselines. We compare our model with two state-of-the-art baselines for category-level novel-view synthesis and 3D reconstruction: CodeNeRF [21] and EG3D [5]. We adapted the latter to the reconstruction task from its original GAN setting for fair comparison. Note that code and data for AutoRF [32], an otherwise relevant baseline, has not been released. Another two baselines, "Ours-E" and "Ours-E-SG", are evaluated. The suffix 'E' denotes an entangled model, that is, a single-stream triplane decoder with density and color heads. The suffix 'SG' denotes the use of a StyleGAN backbone. We focus on four strong baselines to avoid prohibitively expensive training.

Implementation details. Our triplane decoders have 6 upsampling blocks with a maximum width of 648. The density, diffuse, and specular networks are implemented as MLPs with 0/1/1 hidden layers of dimension 64/128/128, with a 5-frequency IDE encoding [48] on the view directions. All latent codes have 256 parameters, except the directional code which has an additional 32 per-frame parameters. We sample 256 points per ray, 256 rays per image, 8 images per instance, and 4 instances per batch. We optimize the network with AdamW [24, 28] with initial learning rates of $5{\times}10^{-5}$, $5{\times}10^{-4}$, and $2.5{\times}10^{-3}$ for the model, camera, and latent code parameters respectively, training for 1M iterations on 4 GPUs, with hyperparameters $\lambda^{\text{m}} = 1$ and

Table 1. Co-reconstruction results on test frames of the NuScenes car dataset [4]. We report the LPIPS distance and the peak signal-to-noise ratio (PSNR) between the estimated and ground-truth masked images, the intersection-over-union (IoU) of the estimated and ground-truth masks, and the Fréchet Inception Distance (FID).

Method	LPIPS ↓	PSNR ↑	IoU ↑	FID ↓
CodeNeRF [21]	0.334	20.3	0.978	117.7
EG3D [5]	0.328	20.5	**0.981**	106.3
Ours-E-SG	0.307	21.6	**0.981**	60.8
Ours-E	0.300	21.5	**0.981**	54.7
Ours	**0.287**	**21.7**	**0.981**	**40.8**

Table 2. Co-reconstruction results on test frames of the ShapeNet dataset [7], for view *extrapolation* (train and test frames from different half-spaces). Where the ground-truth distance maps are provided, we report the mean absolute distance error (D-MAE). Star ($\star$) denotes that training diverged.

	Method	LPIPS ↓	PSNR ↑	IoU ↑	FID ↓	D-MAE ↓
Cars	CodeNeRF [21]	0.140	22.2	0.967	142	0.0316
	EG3D [5]	$\star$	$\star$	$\star$	$\star$	$\star$
	Ours-E-SG	0.120	24.0	0.979	100	0.0245
	Ours-E	**0.117**	**24.1**	**0.980**	**75.9**	0.0220
	Ours	**0.117**	23.7	**0.980**	79.4	**0.0210**
Chairs	CodeNeRF [21]	0.0519	25.4	0.909	12.3	–
	EG3D [5]	$\star$	$\star$	$\star$	$\star$	$\star$
	Ours-E-SG	0.0393	27.8	**0.919**	7.1	–
	Ours-E	**0.0292**	**29.7**	0.910	**4.2**	–
	Ours	0.0319	28.9	0.911	4.9	–

$\lambda^{d} = 0.1$. Complete implementation details are reported in the appendix, and the code will be publicly released.

4.2. Experiments

Co-reconstructing NuScenes cars. For this experiment, we train on the 4157 training instances and evaluate on the reserved test frames. This assesses the ability to interpolate between nearby views, the standard mode for evaluating novel view synthesis algorithms. Note that only a small range of views are seen for the vast majority of instances in this dataset, making the single-instance reconstruction task quite ill-posed. However, a more complete reconstruction is possible if we reconstruct multiple instances together.

The quantitative results are shown in Tab. 1. They indicate that our model can reconstruct the training data with high fidelity, despite sharing almost all parameters between the instances, and can accurately interpolate between nearby views. In particular, we outperform the baseline models with respect to perceptual similarity (LPIPS), since our renders are less blurry and perceptually closer to the ground truth. However, these co-reconstruction models are better evaluated in the extrapolation setting, for viewpoints beyond the cone of training views. This is not possible to evaluate quantitatively on the NuScenes dataset, because the cars are only seen from one side, but can be demonstrated qualitatively by rendering the unseen side of the cars, as shown in Figs. 4 and 5. It is clear that the model is able to learn useful shape and appearance priors, especially a prior on the symmetry of cars, in order to reconstruct effectively across the category. Note the artifacts in the single instance reconstruction examples (floating regions of non-zero density), where the visual evidence was inadequate for correcting the density field, unlike in the co-reconstruction model.

Table 3. Ablation study on the test frames of the NuScenes dataset. Different forms of conditioning and disentangling (S–A: Shape–Appearance; D–S: Diffuse–Specular) are compared.

Conditioning			Disentangling		LPIPS	PSNR	IoU	FID
Concat.	AdaIN	Attn	S–A	D–S	↓	↑	↑	↓
✓					0.334	20.3	0.978	117.7
	✓				0.307	21.6	**0.981**	60.8
		✓			0.300	21.5	**0.981**	54.7
		✓	✓		0.302	21.2	0.980	53.9
		✓	✓	✓	**0.287**	**21.7**	**0.981**	**40.8**

Co-reconstructing ShapeNet categories. Here we assess view *extrapolation* performance, where the test frames are sampled from a different half-space than the train frames, on synthetic (non-incidental) data. Since this dataset has exact camera poses, white backgrounds, static objects, and constant camera intrinsics, we disable camera optimization, mask loss, and view codes. The results are shown in Tab. 2 and Fig. 6, where we evaluate on a subset of 1000 instances ($\sim$41%). In contrast to CodeNeRF, the extrapolated views are more plausible, less blurry, and the geometric error is significantly lower. The improvement is even more significant for the chairs dataset, although disentanglement is slightly detrimental here since the textures are very simple.

Novel instance reconstruction. Here we use the pre-trained model from the previous section, and test its ability to assist in the reconstruction of novel instances. As outlined in Sec. 3, we optimize the cameras and latent codes on a set of 5 training images from each test instance. The qualitative results in Fig. 7 indicate that our model is able to fit to new instances and generate plausible reconstructions.

Ablation study. To investigate the effect of our design choices, we ablate our model in Tab. 3. Cross-attention conditioning and full disentanglement outperform the other approaches, with a marked improvement in realism (FID) attributable to both of these factors.

Visualizing the latent space. A side advantage of having a disentangled model is that we are able to manipulate the reconstruction results in predictable ways. In Fig. 8, we swap in different appearance codes, while keeping the shape code constant, and vice versa. This provides evidence that a disentangled latent space has been learned correctly.

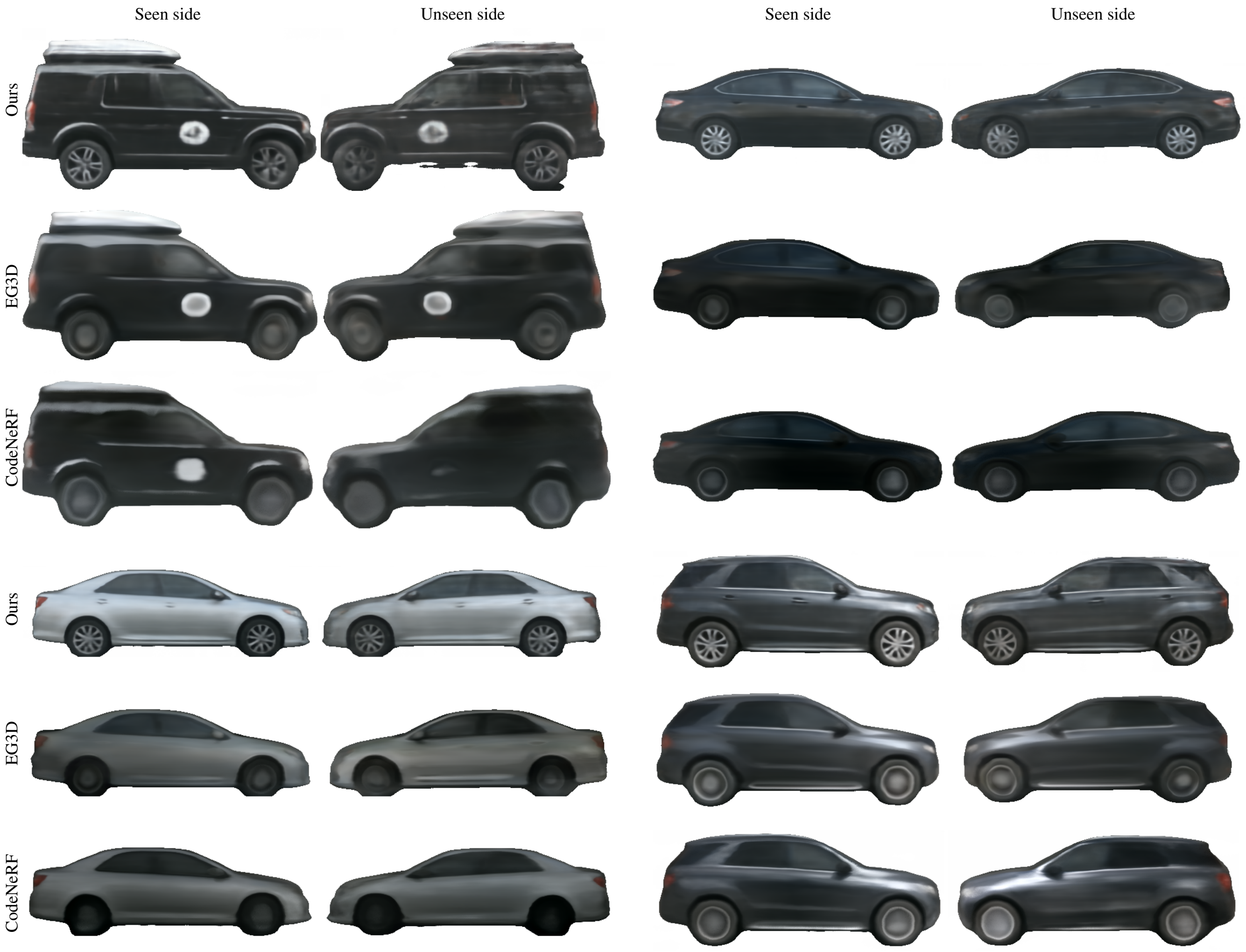


Figure 4. Comparison of co-reconstruction methods, displaying the seen and unseen sides of the cars. Our model produces sharper reconstructions than EG3D [5] and CodeNeRF [21], which is particularly noticeable at the wheels and handles.

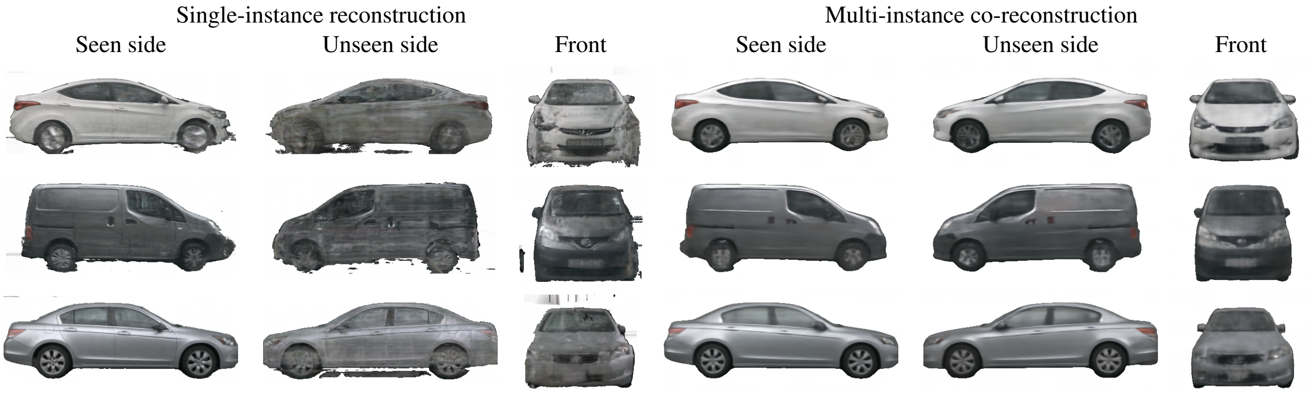

Single-instance reconstruction
Seen side
Unseen side
Front
Multi-instance co-reconstruction
Seen side
Unseen side
Front

Figure 5. Comparison of single-instance reconstruction (left) with multi-instance co-reconstruction (right) for our model, displaying the seen and unseen sides of the cars (interpolation vs extrapolation). While NeRF-like reconstruction can render the seen side plausibly, the co-reconstruction model can extrapolate by learning shape and appearance priors, such as symmetry, smoothness, and part colors.

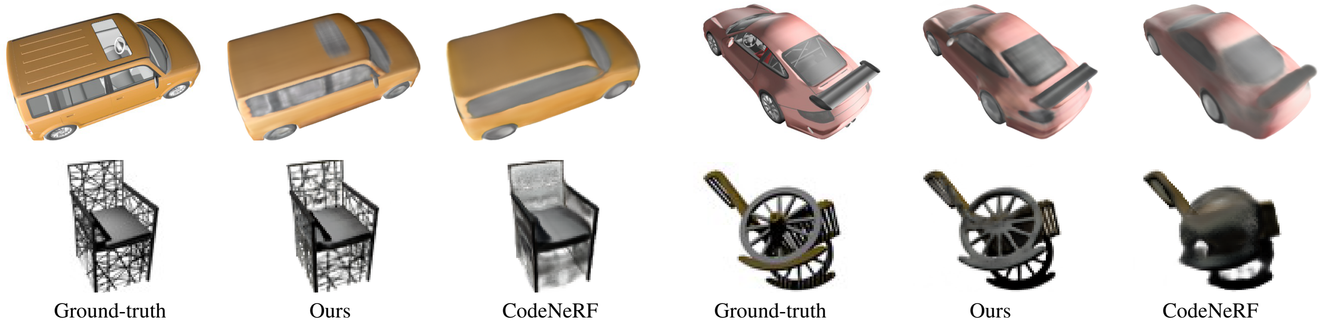


Figure 6. Co-reconstruction results on the ShapeNet dataset. We display the ground-truth image alongside the rendered images from our model and CodeNeRF [21], for test frames from the *unseen* side of the car or chair, testing the ability of the models to extrapolate and learn shape and appearance priors. Our model produces noticeably sharper and more detailed reconstructions than CodeNeRF [21].

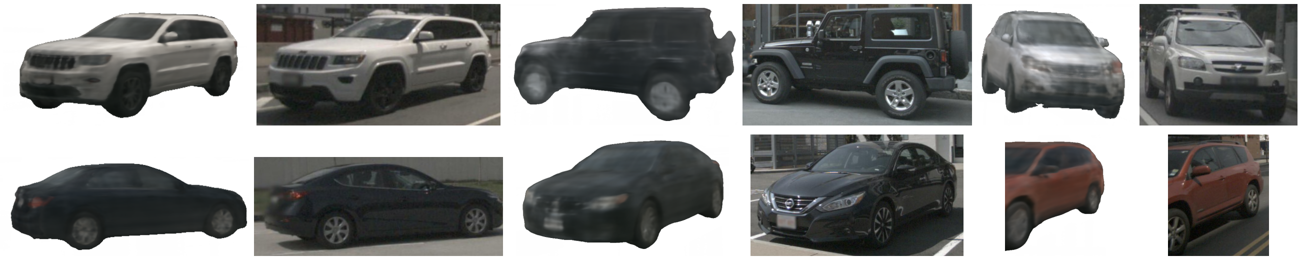


Figure 7. Qualitative results for novel instance reconstruction from 5 input images. The overall shape and texture is recovered, despite the model never having seen the instances during training.

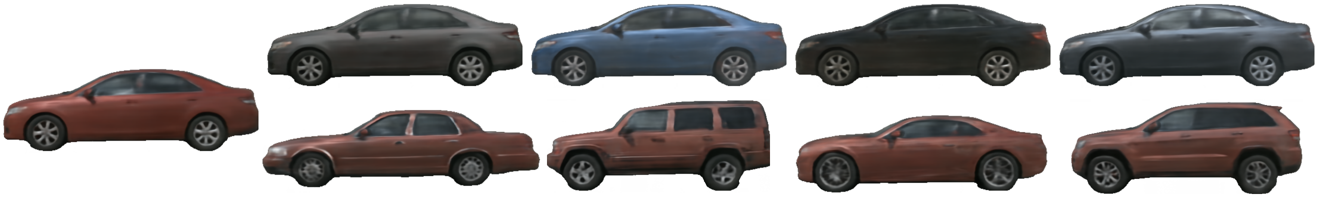


Figure 8. Swapping in different appearance codes for a constant shape code (top) and vice versa (bottom). The original instance is on the left. The learned latent space disentangles shape and appearance effectively, allowing us to manipulate them in isolation.

5. Discussion and Conclusion

Limitations. One limitation of our model is that it is likely to be most beneficial for objects with significant structure, especially those with symmetries. In contrast, the model may learn weaker priors for less structured objects like trees, since their geometries are less predictable. Another limitation is that the model requires approximate instance segmentations for the category, so a pre-trained segmentation network is needed. While many high-quality segmenters are available for vehicles, they may be less accessible or lower quality for other categories. Finally, the latent code memory requirements scale linearly with the number of instances, as does the training time. While this is not excessive, an autoencoding approach may be more appropriate as the dataset size is scaled up further.

Conclusion. We have proposed a method for 3D reconstruction and novel-view synthesis that learns shape and appearance priors from glimpses of the real world. We use a co-reconstruction setting to learn these priors for a single category and demonstrate that the model learns helpful geometric and visual cues, such as symmetries, smoothness, and part colors, which cannot be derived from a single instance in isolation. We believe that this setting is best suited for scaling up the learning of reconstruction priors, because, despite their limited range of views, casual recordings of the real world are a plentiful source of multi-view observations.

Acknowledgements. We are grateful for support from Continental AG, the European Research Council Starting Grant (ERC-UNION-CoG-101001212), and the Royal Academy of Engineering (RF/201819/18/163).

References

[1] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-NeRF: a multiscale representation for anti-aliasing neural radiance fields. In *ICCV*, pages 5855–5864, 2021. 2

[2] Sai Bi, Zexiang Xu, Pratul Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Neural reflectance fields for appearance acquisition. *arXiv preprint arXiv:2008.03824*, 2020. 2

[3] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. NeRD: neural reflectance decomposition from image collections. In *ICCV*, pages 12684–12694, 2021. 2

[4] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. Nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, pages 11621–11631, 2020. 1, 5, 6

[5] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *CVPR*, pages 16123–16133, 2022. 2, 3, 4, 5, 6, 7

[6] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. Pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *CVPR*, pages 5799–5809, 2021. 2

[7] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 5, 6

[8] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. TensoRF: Tensorial radiance fields. In *ECCV*, 2022. 2

[9] Wenzheng Chen, Joey Litalien, Jun Gao, Zian Wang, Clement Fuji Tsang, Sameh Khamis, Or Litany, and Sanja Fidler. DIB-R++: learning to predict lighting and material with a hybrid differentiable renderer. In *NeurIPS*, volume 34, 2021. 2

[10] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 5

[11] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 1, 5

[12] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. In *ICLR*, 2017. 2

[13] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *CVPR*, pages 5501–5510, 2022. 2

[14] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. GET3D: A generative model of high quality 3d textured shapes learned from images. In *NeurIPS*, 2022. 2, 4

[15] Shubham Goel, Angjoo Kanazawa, and Jitendra Malik. Shape and viewpoint without keypoints. In *ECCV*, pages 88–104. Springer, 2020. 2

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, pages 2672–2680, 2014. 2

[17] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4D: Around the World in 3,000 Hours of Egocentric Video. In *CVPR*, 2022. 1

[18] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NeurIPS*, volume 30, 2017. 5

[19] John Houston, Guido Zuidhof, Luca Bergamini, Yawei Ye, Long Chen, Ashesh Jain, Sammy Omari, Vladimir Iglovikov, and Peter Ondruska. One thousand and one hours: Self-driving motion prediction dataset. In *Conference on Robot Learning*, pages 409–418. PMLR, 2021. 5

[20] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, pages 1501–1510, 2017. 2

[21] Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *ICCV*, pages 12949–12958, 2021. 2, 5, 6, 7, 8

[22] Angjoo Kanazawa, Shubham Tulsiani, Alexei A Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, pages 371–386, 2018. 2

[23] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR*, pages 8110–8119, 2020. 2

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[25] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *NeurIPS*, volume 25. Curran Associates, Inc., 2012. 4

[26] Xueting Li, Sifei Liu, Kihwan Kim, Shalini De Mello, Varun Jampani, Ming-Hsuan Yang, and Jan Kautz. Self-supervised single-view 3d reconstruction via semantic consistency. In *ECCV*, pages 677–693. Springer, 2020. 2

[27] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *ACM Transactions on Graphics*, 38(4):1–14, 2019. 2

[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 5

[29] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *CVPR*, pages 7210–7219, 2021. 2

[30] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: representing scenes as neural radiance fields for view synthesis. In *ECCV*, pages 405–421. Springer, 2020. 1, 2, 3

[31] Tom Monnier, Matthew Fisher, Alexei A. Efros, and Mathieu Aubry. Share with thy neighbors: Single-view reconstruction by cross-instance consistency. In *ECCV*, 2022. 2

[32] Norman Müller, Andrea Simonelli, Lorenzo Porzi, Samuel Rota Bulò, Matthias Nießner, and Peter Kontschieder. AutoRF: Learning 3d object radiance fields from single view observations. In *CVPR*, Jun 2022. 2, 5

[33] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, Jul 2022. 2

[34] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *CVPR*, pages 11453–11464, 2021. 2

[35] Daniel Rebain, Mark J Matthews, Kwang Moo Yi, Gopal Sharma, Dmitry Lagun, and Andrea Tagliasacchi. Attention beats concatenation for conditioning neural fields. *arXiv preprint arXiv:2209.10684*, 2022. 3

[36] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *ICCV*, pages 10901–10911, 2021. 1, 2

[37] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 3

[38] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proc. MICCAI*, pages 234–241. Springer, 2015. 3

[39] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *NeurIPS*, 33:20154–20166, 2020. 2

[40] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *NeurIPS*, 33:7462–7473, 2020. 2

[41] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *NeurIPS*, volume 32, 2019. 1, 2, 5

[42] Ivan Skorokhodov, Sergey Tulyakov, Yiqun Wang, and Peter Wonka. EpiGRAF: Rethinking training of 3d gans. In *NeurIPS*, 2022. 2, 4

[43] Pratul P Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron. NeRV: neural reflectance and visibility fields for relighting and view synthesis. In *CVPR*, pages 7495–7504, 2021. 2

[44] Ayush Tewari, Ohad Fried, Justus Thies, Vincent Sitzmann, Stephen Lombardi, Kalyan Sunkavalli, Ricardo Martin-Brualla, Tomas Simon, Jason Saragih, Matthias Nießner, et al. State of the art on neural rendering. In *Computer Graphics Forum*, volume 39, pages 701–727. Wiley Online Library, 2020. 2

[45] Ayush Tewari, Xingang Pan, Ohad Fried, Maneesh Agrawala, Christian Theobalt, et al. Disentangled3D: Learning a 3d generative model with disentangled geometry and appearance from monocular images. In *CVPR*, pages 1516–1525, 2022. 2

[46] A. Tewari, J. Thies, B. Mildenhall, P. Srinivasan, E. Tretschk, W. Yifan, C. Lassner, V. Sitzmann, R. Martin-Brualla, S. Lombardi, T. Simon, C. Theobalt, M. Nießner, J. T. Barron, G. Wetzstein, M. Zollhöfer, and V. Golyanik. Advances in neural rendering. *Computer Graphics Forum*, 41(2):703–735, 2022. 2

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, volume 30, 2017. 3

[48] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *CVPR*, pages 5481–5490. IEEE, 2022. 2, 4, 5

[49] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. IBRNet: learning multi-view image-based rendering. In *CVPR*, pages 4690–4699, 2021. 2

[50] Shangzhe Wu, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. Dove: Learning deformable 3d objects by watching videos. *arXiv preprint arXiv:2107.10844*, 2021. 2

[51] Yufei Ye, Shubham Tulsiani, and Abhinav Gupta. Shelf-supervised mesh prediction in the wild. In *CVPR*, 2021. 2

[52] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. PixelNeRF: neural radiance fields from one or few images. In *CVPR*, pages 4578–4587, 2021. 2

[53] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. NeRF++: analyzing and improving neural radiance fields. *arXiv:2010.07492*, 2020. 2

[54] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 5

[55] Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics*, 40(6):1–18, 2021. 2