Unsupervised Causal Generative Understanding of Images

Titas Anciukevičius [1 2] **Patrick Fox-Roberts** [1] **Edward Rosten** [1] **Paul Henderson** [3]

Abstract

We present a novel causal generative model for unsupervised object-centric 3D scene understanding that generalizes robustly to out-of-distribution images. This model is trained to reconstruct multi-view images via a latent representation describing the shapes, colours and positions of the 3D objects they show. We then propose an inference algorithm that can infer this latent representation given a single out-of-distribution image as input. We conduct extensive experiments applying our approach to test datasets that have zero probability under the training distribution. Our approach significantly out-performs baselines that do not capture the true causal image generation process.

1. Introduction

Most machine learning approaches make the assumption that at test time, they are applied to data drawn from the same distribution as seen during training (Bishop, 2006). This means the powerful generalization guarantees of statistical learning theory apply (Vapnik, 1991). Indeed, most learning-based methods for computer vision do *not* generalize to observations that are statistically different from those seen in their training set – recent works have demonstrated this for images taken from unfamiliar viewpoints (Alcorn et al., 2019; Barbu et al., 2019), shifted by few pixels (Azulay & Weiss, 2019), and showing scenes with an unseen composition of objects (Beery et al., 2018; Ribeiro et al., 2016; Schott et al., 2021; Geirhos et al., 2020; 2019; Dijk & Croon, 2019). It has been suggested that this is because they learn spurious correlations – or *shortcuts* (Geirhos et al., 2020) – to achieve low training loss, but which do not capture the true *causal* relationships that remain universally valid.

In this work, we consider the task of transforming a single observed image into a detailed representation of the scene it depicts, providing explicit information about its 3D structure such as object locations, shapes and appearances. We focus on the setting where at test time, we see images depicting scenes that have *zero* probability in the training distribution (Fig. 1). We also adopt an *unsupervised* approach to learning, avoiding the need for expensive manual annotation of object masks, 3D positions, and similar.

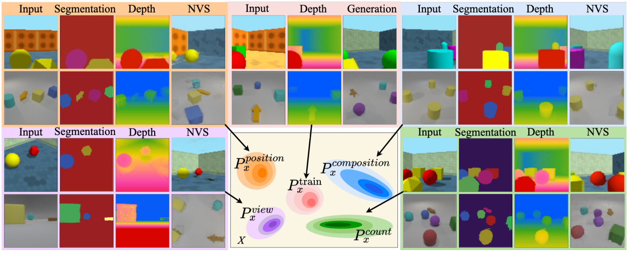


Figure 1. Our generative model is trained to reconstruct multi-view images drawn from its training distribution (red), via a latent object-centric 3D scene representation. After training, it can sample plausible scenes containing multiple objects ('Generation', top center). At test time, it inputs single images drawn from different distributions (e.g. with novel compositions of objects), but is still able to infer segmentations, depth-maps, and novel views.

Our approach is to build a generative model (Sec. 3) jointly over image pixels and the 3D world they depict, whose structure reflects the underlying causal model of the environment (Pearl, 2009), that can be robustly inverted to infer the latent factors that gave rise to an input image. We design its conditional independencies to match those present in the environment, corresponding to *independent mechanisms* (Schölkopf et al., 2012; Janzing & Schölkopf, 2010) in the world. Thus when certain mechanisms or physical processes in the environment change, much of our model remains applicable; this has been called *sparse mechanism shift* (Schölkopf et al., 2021).

In general it is impossible to recover such a causal model purely from observational data (Pearl, 2009). We therefore embed in our model universal knowledge that the world is composed of 3D objects of different shapes, appearing at different locations, imaged by a camera subject to the laws of 3D geometry and perspective projection. Each object is associated with disentangled latent variables describing its position and appearance, allowing the same object to be represented invariantly in different locations (unlike spatial mixture models (Engelcke et al., 2020; Stelzner et al., 2021) which cannot perform inference on scenes with objects at unseen positions because their appearance and position rep-

[1] Snap Inc. [2] University of Edinburgh [3] University of Glasgow . Correspondence to: Titas Anciukevičius <titas.anciukevicius@gmail.com>.

Published at the ICML 2022 Workshop on Spurious Correlations, Invariance, and Stability. Baltimore, Maryland, USA. Copyright 2022 by the author(s).

resentations are entangled (Montero et al., 2022; Locatello et al., 2019)). We perform inference on OOD data by *intervening* on our model (Pearl, 2009) – replacing conditional distributions (or mechanisms) that are no longer appropriate.

Targeting OOD generalization imposes several technical constraints on our model compared with other recent work summarised in Sec. 2, none of which address this setting. Specifically, we must ensure that the inference method itself supports OOD generalization and we cannot employ amortized variational inference (Kingma & Welling, 2014), as an encoder network is in general not robust to changes of distribution (Montero et al., 2022; Geirhos et al., 2020). Instead, we develop a novel Markov chain Monte-Carlo (MCMC) inference scheme, that finds posterior samples for a given test image. We also cannot include a learnt neural renderer (mapping features to pixels in image space) (Niemeyer & Geiger, 2021; Castrejon et al., 2022; Nguyen-Phuoc et al., 2019) in the generative model, as these are not guaranteed to generalise OOD.

We evaluate our model (Sec. 4) with test datasets that have zero probability under their training distribution. We show that our model can generalize to unseen numbers of objects, unseen compositions, and new camera viewpoints, significantly better than existing works, which are susceptible to spurious correlations in the training data.

To summarise, our core contribution is **the first unsupervised framework for inference of explicit object-centric 3D scene representations, that generalizes to out-of-distribution scenes**. Our secondary contributions are: (i) a novel generative model of 3D scenes based on multi-object radiance fields with explicit object positions and volumetric rendering; and (ii) a novel MCMC inference scheme exploiting the structure of our model, that allows inferring 3D scenes from a single out-of-distribution image.

2. Related Work

Recent works have observed that most learning-based computer vision methods fail to generalise on OOD data, and have analysed this (Alcorn et al., 2019; Azulay & Weiss, 2019; Beery et al., 2018; Ribeiro et al., 2016; Schott et al., 2021; Geirhos et al., 2020; 2019; Nagarajan et al., 2021; Milbich et al., 2021; Kortylewski et al., 2021; Yuille & Liu, 2021) as well as constructing various benchmarks (Lake et al., 2017; Barbu et al., 2019; Gulrajani & Lopez-Paz, 2021; Karazija et al., 2021; Ye et al., 2021; Wiles et al., 2022). Others have tried to improve OOD generalisation (Arjovsky et al., 2019; Shi et al., 2021; Ahuja et al., 2020; 2021; Parascandolo et al., 2020; Krueger et al., 2021; Shahtalebi et al., 2021; Liu et al., 2021a). Unlike us, these methods assume that multiple differently-distributed datasets are available during training, and only address the supervised setting.

Our work is also connected to the *vision as inverse graphics* paradigm (Grenander, 1976; Barlow, 1987; Romaszko et al., 2017). In this setting, it is assumed that we have access to (maybe parametric) 3D models of objects, and wish to find suitable pose and other parameters to explain an input image (Loper & Black, 2014; Kulkarni et al., 2015; Jampani et al., 2015; Romaszko et al., 2017; 2020; Izadinia et al., 2017). Like our work, these typically use a test-time optimisation; unlike ours, they assume known priors on object layout and shapes.

Neural implicit scene representations (Tewari et al., 2020) aim to a learn continuous representation of a 3D scene from 2D images using neural rendering, either by explicit volumetric rendering (Mildenhall et al., 2022; Martin-Brualla et al., 2021; Wizadwongsa et al., 2021; Yariv et al., 2020; Niemeyer et al., 2020) or with CNN post-processing of rendered features (Sitzmann et al., 2019; 2020). These initial works fitted individual scenes without learning characteristics common across them, therefore requiring many images as input. This was addressed by sharing models across different scenes (Yu et al., 2021; Trevithick & Yang, 2020; Li et al., 2021; Eslami et al., 2018; Peng et al., 2020; Niemeyer et al., 2021; Jang & Agapito, 2021), allowing inference of novel views from one or few images.

All these methods model a scene as a single monolithic entity, without decomposing it into individual objects. In contrast, (Ost et al., 2021) divides a scene into objects, but requires detailed manual annotations to do so; (Driess et al., 2022) relies on ground-truth object masks. (Yu et al., 2022; Stelzner et al., 2021) discover such a decomposition automatically (though with depth supervision for (Stelzner et al., 2021)). However their entanglement of latent position and appearance means they are not guaranteed to generalise to OOD combinations of position and appearance (Locatello et al., 2019). Finally, (Guo et al., 2020) supports composition of multi-object scenes using neural scattering functions – but these must be learnt from multiple views of single objects, a form of weak supervision.

Other approaches have extended neural rendering to the generative setting (Schwarz et al., 2020; Niemeyer & Geiger, 2021; Nguyen-Phuoc et al., 2019; 2020; Kosiorek et al., 2021; Chan et al., 2021; Devries et al., 2021; Deng et al., 2021), allowing sampling scenes *a priori*. However, these do not allow us to perform inference in the OOD setting.

There are object-centric generative models that can sample plausible images and perform inference, but only in 2D, and therefore cannot support 3D tasks such as reconstruction depth prediction. Some use a full-image spatial mixture model (Engelcke et al., 2020; Nanbo et al., 2020; Kobayashi et al., 2022; Emami et al., 2021; Engelcke et al., 2021; Kabra et al., 2021) or alpha stacking (van Steenkiste et al., 2020; von Kügelgen et al., 2020); others model images as composed of smaller patches or sprites (Eslami et al., 2016; Jiang & Ahn, 2020; Anciukevičius et al., 2020); others use compositional energy-based models (Du et al., 2021; Liu et al., 2021b). A 3D extension of these latter is proposed by (Henderson & Lampert, 2020), using a voxel representation, but this requires videos and does not support OOD generalisation.

3. Method

Our goal is to infer an explicit object-centric representation of a 3D scene from a single image, even when it shows a scene lying outside the distribution observed during training. We build a compositional generative causal model jointly over multi-view images and the scenes they depict (Sec. 3.1). For training (Sec. 3.2) we use only posed multi-view images $(\mathbf{x}_1, \mathbf{v}_1), ..., (\mathbf{x}_N, \mathbf{v}_N)$, without any annotations such as depth-maps, bounding-boxes or segmentation masks. At test time, input images are drawn from another distribution disjoint from the training distribution. Inference over the generative model (Sec. 3.3) yields a object-centric representation, including 3D shapes, positions and appearances.

3.1. Compositional Generative Causal Model

We model a multi-view set of N images $\{\mathbf{x}_1 \ldots \mathbf{x}_N\}$ as caused by a single 3D scene $\mathcal{S}$ being rendered from viewpoints $\{\mathbf{v}_1 \ldots \mathbf{v}_N\}$, by a function $C(\mathcal{S}, \mathbf{v})$. We now describe this scene representation, then the generative process by which it is sampled.

The scene $\mathcal{S} = \left(\mathbf{s}_{bg}, \{(\mathbf{s}_i^{app}, \mathbf{s}_i^{pos})\}_{i=1}^{O}\right)$ is composed of a 3D background component $\mathbf{s}_{bg}$ describing the background's shape and color, and 3D objects indexed $i = 1 \ldots O$ with shape and color described by $\mathbf{s}_i^{app}$ and explicit 3D positions $\mathbf{s}_i^{pos}$. Each $\mathbf{s}_i^{app}$ explicitly represents the 3D appearance of an object as a neural radiance field (NeRF) (Mildenhall et al., 2022), placed in a *canonical space* (e.g. with the object centered at the origin). The positions $\mathbf{s}_i^{pos}$ specify where the objects are placed in a global 3D *scene space*. In contrast to prior work that parametrized 3D object positions as coordinates, we use 1-hot vectors choosing from a set of candidate locations at which to center the object.

Generative process for $\mathcal{S}$. We first sample a high-level latent scene embedding $\mathbf{z}^g \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, that will model correlations between objects and learn the typical composition of a scene (Jiang & Ahn, 2020; Anciukevičius et al., 2020). The individual object appearances are specified by latent $\mathbf{z}_i^{shape}$ and $\mathbf{z}_i^{col}$ that respectively encode the shape and color of the i^{th} object; they are conditioned on $\mathbf{z}^g$ and given by a fully-connected network $\zeta_\theta(\mathbf{z}^g)$ with weights θ. The position $\mathbf{z}_i^{pos}$ is specified by a categorical variable, with logits given by $\xi_\theta(\mathbf{z}^g)$. We similarly introduce latents $\mathbf{z}_{bg}^{shape}$ and $\mathbf{z}_{bg}^{col}$ to encode the shape and color of the background. For brevity, we will write $\mathbf{z}^s = \{\mathbf{z}_{bg}^{shape}, \mathbf{z}_{bg}^{col}, \mathbf{z}_{1 \ldots O}^{shape}, \mathbf{z}_{1 \ldots O}^{col}, \mathbf{z}_{1 \ldots O}^{pos}\}$. The latents $\mathbf{z}^s$ are mapped to scene $\mathcal{S}$ by a function S_θ. This sets $\mathbf{s}_i^{pos}$ equal to $\mathbf{z}_i^{pos}$, and derives object NeRF representations $\mathbf{s}_i^{app}$ from $\mathbf{z}_i^{shape}$ and $\mathbf{z}_i^{col}$ as described in the next paragraph. The probability of image $\mathbf{x}_n$ at viewpoint $\mathbf{v}_n$ is

$$p_\theta(\mathbf{x}_n \,|\, \mathbf{v}_n) = \iint f_{\mathcal{N}}(\mathbf{x}_n; C(S_\theta(\mathbf{z}^s), \mathbf{v}_n), \sigma^2) p_\theta(\mathbf{z}^s \,|\, \mathbf{z}^g) p_\theta(\mathbf{z}^g) \mathrm{d}\mathbf{z}^s \mathrm{d}\mathbf{z}^g \tag{1}$$

where $C(\mathcal{S}, \mathbf{v})$ renders the scene described by $\mathcal{S}$ from viewpoint $\mathbf{v}$, and $f_{\mathcal{N}}$ represents a factored Gaussian likelihood over the $H \times W \times 3$ pixels of the image, with fixed standard deviation σ. The probability of a composition $\mathbf{z}^s$ of objects and background in a scene is given by

$$p_\theta(\mathbf{z}^s \,|\, \mathbf{z}^g) = p_\theta(\mathbf{z}_{bg}^{shape}, \mathbf{z}_{bg}^{col} \,|\, \mathbf{z}^g) \prod_{i=1}^{O} p_\theta(\mathbf{z}_i^{shape}, \mathbf{z}_i^{col}, \mathbf{z}_i^{pos} \,|\, \mathbf{z}^g) \tag{2}$$

Rendering the scene $\mathcal{S}$. The rendering process $C(\mathcal{S}, \mathbf{v})$ outputs an image $\mathbf{x}$ for a camera viewpoint $\mathbf{v}$, showing a scene $\mathcal{S}$. Recall $\mathcal{S}$ contains a 3D background component $\mathbf{s}_{bg}$ and a set of object components $\{\mathbf{s}_i^{app},\, \mathbf{s}_i^{pos}\}_{i=1}^{O}$; we will identify the background as component $i = 0$, with $\mathbf{s}_0^{pos}$ fixed to the origin. We extend the multi-component neural radiance fields of Martin-Brualla et al. (2021) to support explicit placement of objects in the scene according to position variables $\mathbf{s}_i^{pos}$. Specifically, the latents $\mathbf{z}_i^{shape}$ and $\mathbf{z}_i^{col}$ for the i^{th} object parametrize a learnt function $f_\theta^*(\mathbf{q}^*;\, \mathbf{z}_i^{shape},\, \mathbf{z}_i^{col})$, that maps points $\mathbf{q}^*$ in the canonical space of the object to a color $c \in [0, 1]^3$ and density $\sigma \in \mathbb{R}^+$. We place each object at its 3D position $\mathbf{s}_i^{pos}$ by convolving its density and color functions with a one-hot location indicator:

$$f_i(\mathbf{q}) = \int_{\mathbf{q}^*} s_i^{pos}(\mathbf{q}^*) \cdot f_\theta^*\big(\mathbf{q} - \mathbf{q}^*;\, \mathbf{z}_i^{shape}, \mathbf{z}_i^{col}\big)\, \mathrm{d}\mathbf{q}^* \equiv \big(c_i(\mathbf{q}), \sigma_i(\mathbf{q})\big) \tag{3}$$

where $\mathbf{q}$ is a position in scene space, and $s_i^{pos}(\mathbf{q})$ is an indicator function with a unit impulse if point $\mathbf{q}$ is chosen as the center position of the object by the 1-hot indicator $\mathbf{s}_i^{pos}$. Similar to (Yu et al., 2022), we divide the scene space into foreground and background regions, and only render the corresponding components in each. Given the placed object densities σ_i and colors c_i, we calculate the color of each pixel $C(\mathcal{S},\, \mathbf{v})[\mathbf{r}]$ in the image $\mathbf{x}$ by casting a ray $\mathbf{r}(t) = \mathbf{x_0} + t\mathbf{d} \in \mathbb{R}^3$ from the pixel in direction $\mathbf{d}$ through a camera at position $\mathbf{x_0}$, summing the contributions from different objects (Martin-Brualla et al., 2021; Max, 1995).

Continuous relaxation of object placement. To allow gradient-based training and inference, we relax the categorical position variable to a Gumbel-Softmax (Jang et al., 2017; Maddison et al., 2016). This approach ensures we always receive non-zero gradients of the image with respect to every possible object position, easing optimisation. This is in contrast to models based on spatial transformers (Niemeyer & Geiger, 2021; Xue et al., 2022), which can get stuck in local minima if the model has a poor initial prediction, as the gradient of pixels with respect to position is zero if the predicted and true positions do not overlap.

	GQN							ARROW						
	per-image				per-scene		FID ↓	per-image				per-scene		FID ↓
	PSNR ↑	D.MRE ↓	ARI ↑	mSC ↑	PSNR ↑	D.MRE ↓		PSNR ↑	D.MRE ↓	ARI ↑	mSC ↑	PSNR ↑	D.MRE ↓	
Test														
Ours	24.1	0.031	0.81	**0.88**	20.8	0.058	**80.3**	27.1	**0.100**	**0.71**	**0.82**	26.8	**0.107**	**141.4**
Slot Att.	30.5	–	**0.94**	0.67	–	–	–	28.3	–	0.48	0.17	–	–	–
uORF*	27.0	0.027	0.74	0.59	24.2	0.049	–	**35.1**	0.176	0.64	0.44	**33.8**	0.202	–
NeRF-VAE*	**32.0**	**0.016**	–	–	**27.8**	**0.033**	84.2	25.3	0.991	–	–	25.3	0.991	182.7
OOD														
Ours	**21.8**	**0.034**	**0.68**	**0.89**	**18.3**	**0.069**		**26.7**	0.139	**0.57**	**0.81**	**26.2**	0.137	
Slot Att.	20.3	–	0.66	0.56	–	–		22.8	–	0.26	0.14	–	–	
uORF*	14.7	0.287	0.45	0.45	14.1	0.308		22.7	**0.132**	0.38	0.41	22.6	**0.131**	
NeRF-VAE*	15.9	0.271	–	–	14.9	0.301		19.4	0.992	–	–	19.4	0.992	

Table 1. Quantitative results on all tasks, comparing performance for different methods on an in-distribution test set and OOD data. Dashes indicate the method does not support the task. Best results are shown in **bold**.

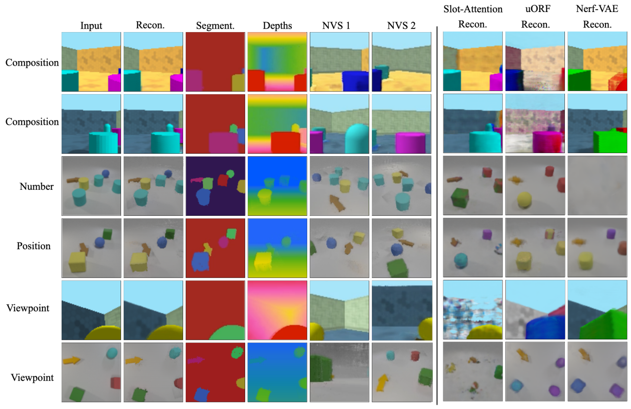


Figure 2. Qualitative results using our model and the baselines on out-of-distribution (OOD) data. Each row shows the input image, then (col. 2–6) outputs from our model: the reconstruction, instance segmentation, depth map, and two novel viewpoints. Our model predicts high-quality segmentations and depth-maps, and new viewpoints that are plausible and consistent with the input. The final three columns show reconstructions (the easiest task) by the baselines. Note that that they fail to generalise to OOD data.

3.2. Training

We train our generative model without any labelled supervision from a dataset of images containing K views for each of T scenes. The model includes three learnable components, with parameters θ: (i) $f^*_\theta(\mathbf{q};\, \mathbf{z}^{shape}, \mathbf{z}^{col})$ that represents a 3D object as a function from position to color and density conditioned on the object appearance embedding; (ii) $f^{bg}_\theta(\mathbf{q};\, \mathbf{z}^{shape}_{bg}, \mathbf{z}^{color}_{bg})$ that similarly represents the 3D background; (iii) ζ_θ and ξ_θ, that map the global scene latent $\mathbf{z}^g$ to parameters of the object and background latents $\mathbf{z}^s$. We train these components using autoencoding variational Bayes (Kingma & Welling, 2014; Rezende et al., 2014). Full details are given in the supplementary (Sec. 7).

3.3. Inference for out-of-distribution (OOD) images

At test time, we assume images are sampled from a distribution disjoint from the training distribution. This means that directly performing posterior inference under our model (which has learnt the training distribution, and ideally assigns zero probability to OOD test images) is not sound. We therefore make appropriate *interventions* on our model (Pearl, 2009), taking advantage of its causal nature. For example, when the distribution of object arrangements is different at test time, we replace the learnt prior $p_\theta(\mathbf{z}^s)$ on object arrangements with an uninformative uniform prior. Moreover, the variational encoder networks enc^s_ϕ and enc^g_ϕ used during training are not suitable for use at test time, due to domain shift in their inputs. Therefore, our framework instead directly samples the posterior distribution of latent variables given an observed image, using Markov chain Monte-Carlo (MCMC) inference.

Our novel MCMC scheme alternates Langevin dynamics (LD) (Besag, 1994; Welling & Teh, 2011) and Metropolis-Hastings (MH) (Hastings, 1970) steps, to infer the latent scene variables $(\mathbf{z}^s, \mathbf{z}^g)$ from a *single* observed image $\mathbf{x}^*$ with viewpoint $\mathbf{v}^*$. The MH steps encourage the Markov chain to make large jumps between modes of the posterior, while the LD steps generate high-probability samples with less exploration. Each LD step ascends the gradient of

$$\log f_{\mathcal{N}}(\mathbf{x}^*;\, C(S_\theta(\mathbf{z}^s),\, \mathbf{v}^*),\, \sigma^2) + \log p_\theta(\mathbf{z}^s \mid \mathbf{z}^g) + \log p(\mathbf{z}^g) \propto \log p(\mathbf{z}^g,\, \mathbf{z}^s \mid \mathbf{x}^*,\, \mathbf{v}^*) \qquad (4)$$

Each MH step first picks an object slot i uniformly at random. It then samples new latents for that object from a proposal distribution $\tilde{p}(\mathbf{z}^{shape}_i,\, \mathbf{z}^{col}_i)$, and decides whether to accept this transition according to the usual Metropolis-Hastings acceptance criterion (Hastings, 1970). The proposal distribution $\tilde{p}$ approximates $\frac{1}{J}\sum_{i=1}^{J} p(\mathbf{z}^{shape}_i,\, \mathbf{z}^{col}_i \mid \mathbf{z}_g)$ using a Gaussian mixture model; it thus captures the distribution of object latent codes while disregarding the ordering of object indices. Note that the compositionality of our model increases sampling efficiency of the chain, as we can consider proposals for each object independently – in contrast to a monolithic latent embedding, which would require accepting or rejecting global modifications to the entire scene. We perform ten LD steps, followed by a single MH step, and repeat this overall process until convergence. When evaluating results, we take the posterior mode, i.e. the sample from each chain with maximum probability.

4. Experiments

Datasets. We conduct experiments on two synthetic image datasets that have some spurious relationship between components in the training set: ARROW (Jiang & Ahn, 2020; Johnson et al., 2017), consisting of four objects – two of which are the same, a third that is different, and an arrow which always points at the odd object; and GQN (Eslami et al., 2018) – which has 3-4 objects placed in a room next to odd-textured wall. We first evaluate performance on the test-set distribution which matches the training distribution. Then, we evaluate generalisation to OOD data, by using several OOD test splits for each dataset, which have zero probability under the training distribution (e.g. relationship between components is broken). Detailed descriptions of dataset splits are given in the supplementary (Sec. 10).

Tasks and Metrics. We consider the following tasks for a single input image: instance segmentation, measured by foreground Adjusted Rand Index (**ARI**) and mean segmentation covering (**mSC**) (Engelcke et al., 2020); depth prediction, measured by the mean relative error of the predicted depths (**D.MRE**); and pixel reconstruction, measured by peak signal-noise ratio (**PSNR**). We report results with two protocols: (i) PER-IMAGE, calculating the metrics only on the input image; and (ii) PER-SCENE, calculating the metrics jointly over multiple images of the scene but still having received only one image as input. The latter setting measures how well the model predicts appearance, depth and segmentation from novel viewpoints. Finally, we evaluate image generation, measured by the Fréchet inception distance (**FID**) (Heusel et al., 2017) between sampled and ground-truth images.

Baselines. We compare our approach to three existing works. **Slot Attention** (Locatello et al., 2020) is a recent unsupervised object segmentation model, with a spatial mixture representation. It is a 2D non-generative model but it does support generalisation to differing numbers of objects at test time. **uORF** (Yu et al., 2022) decomposes 2D images into 3D components represented as NeRFs. Unlike ours, it is not generative, and does not explicitly represent position separate from appearance. **NeRF-VAE** (Kosiorek et al., 2021) is a generative method but it does not separate individual objects in its latent space. We reimplemented (Yu et al., 2022; Kosiorek et al., 2021) in our own framework, and denote these as uORF* and NeRF-VAE*.

Results – in-distribution. Results on all tasks are given in Tab. 1; the top four rows show performance on the test split, which is drawn from the same distribution as the training data. Qualitative results are displayed in Fig. 2, and standard deviations on all results are given in the supplementary. We see that per-image, all methods successfully reconstruct input images (high PSNR) and the 3D-aware methods also accurately predict depths (low D.MRE). On GQN NeRF-VAE* slightly out-performs other approaches, while on ARROW uORF* shows the best PSNR, and our method the best D.MRE. A similar pattern holds for novel views (per-scene setting). For segmentation, Slot Attention gets the best ARI score for GQN, and otherwise ours shows better ARI and mSC scores. Qualitative results for generation (Fig. 3) show that our model has learnt the distribution of both datasets, including likely object and background appearances as well as their compositions. Quantitatively on FID (Tab. 1), ours out-performs NeRF-VAE*.

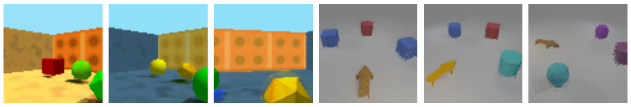
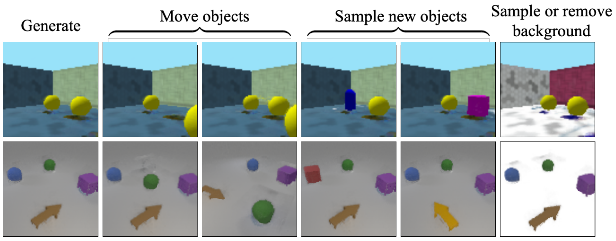


Figure 3. Images sampled from our model for GQN and ARROW

Results – OOD. Quantitative OOD results are given in the bottom half of Tab. 1. This shows the mean over all OOD splits; a full breakdown is given in the supplementary. In general our method significantly out-performs the baselines on out-of-distribution dataset, and performs best on all combinations of tasks and datasets except D.MRE on ARROW. Thus, the best methods on the in-distribution test split are *not* the best on OOD data – absent a causal structure they learn shortcuts (Geirhos et al., 2020) that do not generalise well on the OOD splits. Qualitative results (Fig. 2) reinforce this interpretation – our method predicts accurate segmentations and depth-maps for OOD data, successfully reconstructs the input image via its latent space, and synthesises plausible new viewpoints, in spite of never having seen such an image during training. In contrast, the baselines methods struggle – e.g. mispredicting object colors (1st row, uORF* and NeRF-VAE*), predicting an over-smoothed average scene (3rd row, NeRF-VAE*), or failing to separate walls and ceiling in an unfamiliar pose (5th row, Slot Attention). In Sec. 6, we provide visualise and analyse additional results providing evidence that our model is not susceptible to spurious correlations as compared to non-causal baselines. In the figure below, we also show that our interpretable latent space allows users to edit the scene.



5. Acknowledgments

We thank Professor Christopher K. I. Williams for feedback on the draft and Professor Iasonas Kokkinos and Hakan Bilen for fruitful discussions.

References

Ahuja, K., Shanmugam, K., Varshney, K., and Dhurandhar, A. Invariant risk minimization games. In *International Conference on Machine Learning*, pp. 145–155. PMLR, 2020. 2

Ahuja, K., Caballero, E., Zhang, D., Gagnon-Audet, J.-C., Bengio, Y., Mitliagkas, I., and Rish, I. Invariance principle meets information bottleneck for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 34, 2021. 2

Alcorn, M. A., Li, Q., Gong, Z., Wang, C., Mai, L., Ku, W.-S., and Nguyen, A. Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4845–4854, 2019. 1, 2, 12

Anciukevičius, T., Lampert, C. H., and Henderson, P. Object-centric image generation with factored depths, locations, and appearances. *arXiv preprint arXiv:2004.00642*, 2020. 3, 22

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *CoRR*, abs/1907.02893, 2019. URL `http://arxiv.org/abs/1907.02893`. 2

Azulay, A. and Weiss, Y. Why do deep convolutional networks generalize so poorly to small image transformations? *J. Mach. Learn. Res.*, 20:184:1–184:25, 2019. 1, 2

Barbu, A., Mayo, D., Alverio, J., Luo, W., Wang, C., Gutfreund, D., Tenenbaum, J., and Katz, B. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019. 1, 2

Barlow, H. Cerebral cortex as model builder. In *Matters of intelligence*, pp. 395–406. Springer, 1987. 2

Beery, S., Van Horn, G., and Perona, P. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 456–473, 2018. 1, 2

Besag, J. Comments on "Representations of knowledge in complex systems" by U. Grenander and M.I. Miller. *Journal of the Royal Statistical Society, Series B*, 56:591–592, 1994. 4

Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer, 2006. 1

Burgess, C. P., Matthey, L., Watters, N., Kabra, R., Higgins, I., Botvinick, M. M., and Lerchner, A. Monet: Unsupervised scene decomposition and representation. *CoRR*, abs/1901.11390, 2019. URL `http://arxiv.org/abs/1901.11390`. 22

Castrejon, L., Ballas, N., and Courville, A. INFERNO: Inferring object-centric 3d scene representations without supervision, 2022. URL `https://openreview.net/forum?id=YVa8X_2I1b`. 2

Chan, E. R., Monteiro, M., Kellnhofer, P., Wu, J., and Wetzstein, G. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5799–5809, 2021. 2

Chen, C., Deng, F., and Ahn, S. Roots: Object-centric representation and rendering of 3d scenes. *J. Mach. Learn. Res.*, 22:259:1–259:36, 2021. 19, 22

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1): 1–38, 1977. 19

Deng, Y., Yang, J., Xiang, J., and Tong, X. Gram: Generative radiance manifolds for 3d-aware image generation, 2021. URL `https://arxiv.org/abs/2112.08867`. 2

Devries, T., Bautista, M. Á., Srivastava, N., Taylor, G. W., and Susskind, J. M. Unconstrained scene generation with locally conditioned radiance fields. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14284–14293, 2021. 2

Dijk, T. v. and Croon, G. d. How do neural networks see depth in single images? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2183–2191, 2019. 1

Driess, D., Huang, Z., Li, Y., Tedrake, R., and Toussaint, M. Learning multi-object dynamics with compositional neural radiance fields. *CoRR*, abs/2202.11855, 2022. URL `https://arxiv.org/abs/2202.11855`. 2

Du, Y., Li, S., Sharma, Y., Tenenbaum, B. J., and Mordatch, I. Unsupervised learning of compositional energy concepts. In *Advances in Neural Information Processing Systems*, 2021. 3

Ehrhardt, S., Groth, O., Monszpart, A., Engelcke, M., Posner, I., Mitra, N. J., and Vedaldi, A. RELATE: physically plausible multi-object scene synthesis using structured latent spaces. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL `https://proceedings.`

neurips.cc/paper/2020/hash/806beafe154032a5b818e97b4420ad98-Abstract.html. 22

Emami, P., He, P., Ranka, S., and Rangarajan, A. Efficient iterative amortized inference for learning symmetric and disentangled multi-object representations. In *International Conference on Machine Learning*, pp. 2970–2981. PMLR, 2021. 2

Engelcke, M., Kosiorek, A. R., Jones, O. P., and Posner, I. GENESIS: generative scene inference and sampling with object-centric latent representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020. URL https://openreview.net/forum?id=BkxfaTVFwH. 1, 2, 5, 19, 22

Engelcke, M., Jones, O. P., and Posner, I. GENESIS-V2: inferring unordered object representations without iterative refinement. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 8085–8094, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/43ec517d68b6edd3015b3edc9a11367b-Abstract.html. 2, 22

Eslami, S., Heess, N., Weber, T., Tassa, Y., Szepesvari, D., Hinton, G. E., et al. Attend, infer, repeat: Fast scene understanding with generative models. *Advances in Neural Information Processing Systems*, 29:3225–3233, 2016. 3, 22

Eslami, S. A., Rezende, D. J., Besse, F., Viola, F., Morcos, A. S., Garnelo, M., Ruderman, A., Rusu, A. A., Danihelka, I., Gregor, K., et al. Neural scene representation and rendering. *Science*, 360(6394):1204–1210, 2018. 2, 5, 15, 19, 22

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019. URL https://openreview.net/forum?id=Bygh9j09KX. 1, 2

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020. 1, 2, 5, 12

Greff, K., Kaufman, R. L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M., and Lerchner, A. Multi-object representation learning with iterative variational inference. In *International Conference on Machine Learning*, pp. 2424–2433. PMLR, 2019. 22

Grenander, U. Lectures in pattern theory-volume 1: Pattern synthesis. *Applied Mathematical Sciences*, 1976. 2

Gulrajani, I. and Lopez-Paz, D. In search of lost domain generalization. In *ICLR*, 2021. 2

Guo, M., Fathi, A., Wu, J., and Funkhouser, T. A. Object-centric neural scene rendering. *CoRR*, abs/2012.08503, 2020. URL https://arxiv.org/abs/2012.08503. 2

Hastings, W. K. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57 (1):97–109, April 1970. doi: 10.1093/biomet/57.1.97. 4

Henderson, P. and Lampert, C. H. Unsupervised object-centric video generation and decomposition in 3D. In *Advances in Neural Information Processing Systems (NeurIPS) 33*, 2020. 3, 19, 22

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30, 2017. 5

Higgins, I., Matthey, L., Pal, A., Burgess, C. P., Glorot, X., Botvinick, M. M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl. 15, 19

Hinton, G. Taking inverse graphics seriously, 2013. 12

Hinton, G. E. How to represent part-whole hierarchies in a neural network. *CoRR*, abs/2102.12627, 2021. URL https://arxiv.org/abs/2102.12627. 12

Izadinia, H., Shan, Q., and Seitz, S. M. Im2cad. In *CVPR*, pp. 5134—-5143, 2017. 2

Jampani, V., Nowozin, S., Loper, M., and Gehler, P. V. The informed sampler. *Comput. Vis. Image Underst.*, 136(C): 32–44, 2015. 2

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017. URL https://openreview.net/forum?id=rkE3y85ee. 3

Jang, W. and Agapito, L. Codenerf: Disentangled neural radiance fields for object categories. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 12929–12938. IEEE, 2021. doi: 10.1109/ICCV48922.2021.01271. URL https://doi.org/10.1109/ICCV48922.2021.01271. 2

Janzing, D. and Schölkopf, B. Causal inference using the algorithmic markov condition. *IEEE Trans. on Information Theory*, 2010. 1

Jiang, J. and Ahn, S. Generative neurosymbolic machines. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/94c28dcfc97557df0df6d1f7222fc384-Abstract.html. 3, 5, 15, 20, 22

Jiang, J., Janghorbani, S., de Melo, G., and Ahn, S. SCALOR: generative world models with scalable object representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. URL https://openreview.net/forum?id=SJxrKgStDH. 22

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. CLEVR: a diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017. 5, 20

Kabra, R., Zoran, D., Erdogan, G., Matthey, L., Creswell, A., Botvinick, M. M., Lerchner, A., and Burgess, C. P. Simone: View-invariant, temporally-abstracted object representations via unsupervised video decomposition. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 20146–20159, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/a860a7886d7c7e2a8d3eaac96f76dc0d-Abstract.html. 2, 22

Karazija, L., Laina, I., and Rupprecht, C. Clevrtex: A texture-rich benchmark for unsupervised multi-object segmentation. In Vanschoren, J. and Yeung, S. (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/e2c420d928d4bf8ce0ff2ec19b371514-Abstract-round2.html. 2

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980. 15, 19

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In Bengio, Y. and LeCun, Y. (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6114. 2, 4, 15

Kobayashi, Y., Suzuki, M., and Matsuo, Y. Learning global spatial information for multi-view object-centric models, 2022. URL https://openreview.net/forum?id=3mm5rjb7nR8. 2

Kortylewski, A., He, J., Liu, Q., Cosgrove, C., Yang, C., and Yuille, A. L. Compositional generative networks and robustness to perceptible image changes. In *2021 55th Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–8. IEEE, 2021. 2

Kosiorek, A., Sabour, S., Teh, Y. W., and Hinton, G. E. Stacked capsule autoencoders. *Advances in neural information processing systems*, 32, 2019. 12

Kosiorek, A. R., Strathmann, H., Zoran, D., Moreno, P., Schneider, R., Mokrá, S., and Rezende, D. J. Nerf-vae: A geometry aware 3d scene generative model. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5742–5752. PMLR, 2021. URL http://proceedings.mlr.press/v139/kosiorek21a.html. 2, 5, 20, 22

Krueger, D., Caballero, E., Jacobsen, J.-H., Zhang, A., Binas, J., Zhang, D., Le Priol, R., and Courville, A. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pp. 5815–5826. PMLR, 2021. 2

Kulkarni, T. D., Kohli, P., Tenenbaum, J. B., and Mansinghka, V. Picture: A probabilistic programming language for scene perception. In *CVPR*, pp. 4390—-4399, 2015. 2

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017. 2

Li, J., Feng, Z., She, Q., Ding, H., Wang, C., and Lee, G. H. Nemi: Unifying neural radiance fields with multiplane images for novel view synthesis. *CoRR*, abs/2103.14910, 2021. URL `https://arxiv.org/abs/2103.14910`. 2

Lin, Z., Wu, Y., Peri, S. V., Sun, W., Singh, G., Deng, F., Jiang, J., and Ahn, S. SPACE: unsupervised object-oriented scene representation via spatial attention and decomposition. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020. URL `https://openreview.net/forum?id=rkl03ySYDH`. 22

Liu, C., Sun, X., Wang, J., Tang, H., Li, T., Qin, T., Chen, W., and Liu, T.-Y. Learning causal semantic representation for out-of-distribution prediction. In *NeurIPS*, 2021a. 2

Liu, N., Li, S., Du, Y., Tenenbaum, J., and Torralba, A. Learning to compose visual relations. *Advances in Neural Information Processing Systems*, 34, 2021b. 3

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019. 2

Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., and Kipf, T. Object-centric learning with slot attention. *Advances in Neural Information Processing Systems*, 33:11525–11538, 2020. 5, 12, 21

Loper, M. M. and Black, M. J. Opendr: An approximate differentiable renderer. In *European Conference on Computer Vision*, pp. 154–169. Springer, 2014. 2

Maddison, C. J., Mnih, A., and Teh, Y. W. The concrete distribution: A continuous relaxation of discrete random variables. *CoRR*, abs/1611.00712, 2016. URL `http://arxiv.org/abs/1611.00712`. 3

Martin-Brualla, R., Radwan, N., Sajjadi, M. S., Barron, J. T., Dosovitskiy, A., and 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. 2, 3

Max, N. Optical models for direct volume rendering. *IEEE Trans. on Visualization and Computer Graphics*, 1995. 3

Milbich, T., Roth, K., Sinha, S., Schmidt, L., Ghassemi, M., and Ommer, B. Characterizing generalization under out-of-distribution shifts in deep metric learning. *Advances in Neural Information Processing Systems*, 34, 2021. 2

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. Nerf: representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 65(1):99–106, 2022. doi: 10.1145/3503250. URL `https://doi.org/10.1145/3503250`. 2, 3, 18, 19

Montero, M. L., Bowers, J. S., Costa, R. P., Ludwig, C. J. H., and Malhotra, G. Lost in latent space: Disentangled models and the challenge of combinatorial generalisation. *arXiv preprint arXiv:2204.02283*, 2022. 2

Nagarajan, V., Andreassen, A., and Neyshabur, B. Understanding the failure modes of out-of-distribution generalization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, 2021. URL `https://openreview.net/forum?id=fSTD6NFIW_b`. 2

Nanbo, L., Eastwood, C., and Fisher, R. B. Learning object-centric representations of multi-object scenes from multiple views. In *34th Conference on Neural Information Processing Systems*, 2020. 2, 20, 22

Nguyen-Phuoc, T., Li, C., Theis, L., Richardt, C., and Yang, Y.-L. Hologan: Unsupervised learning of 3d representations from natural images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7588–7597, 2019. 2

Nguyen-Phuoc, T., Richardt, C., Mai, L., Yang, Y.-L., and Mitra, N. Blockgan: Learning 3d object-aware scene representations from unlabelled images. *arXiv preprint arXiv:2002.08988*, 2020. 2

Niemeyer, M. and Geiger, A. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11453–11464, 2021. 2, 3, 18, 22

Niemeyer, M., Mescheder, L., Oechsle, M., and Geiger, A. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3504–3515, 2020. 2

Niemeyer, M., Barron, J. T., Mildenhall, B., Sajjadi, M. S. M., Geiger, A., and Radwan, N. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. *CoRR*, abs/2112.00724, 2021. URL `https://arxiv.org/abs/2112.00724`. 2

Ost, J., Mannan, F., Thuerey, N., Knodt, J., and 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. 2

Parascandolo, G., Neitz, A., Orvieto, A., Gresele, L., and Schölkopf, B. Learning explanations that are hard to vary. *arXiv preprint arXiv:2009.00329*, 2020. 2

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`. 19

Pearl, J. *Causality*. Cambridge University Press, 2 edition, 2009. doi: 10.1017/CBO9780511803161. 1, 2, 4

Peng, S., Niemeyer, M., Mescheder, L., Pollefeys, M., and Geiger, A. Convolutional occupancy networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pp. 523–540. Springer, 2020. 2

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pp. 1278–1286. PMLR, 2014. 4, 15

Ribeiro, M. T., Singh, S., and Guestrin, C. "why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016. 1, 2

Romaszko, L., Williams, C. K., Moreno, P., and Kohli, P. Vision-as-inverse-graphics: Obtaining a rich 3d explanation of a scene from a single image. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 851–859, 2017. 2

Romaszko, L., Williams, C. K., and Winn, J. Learning direct optimization for scene understanding. *Pattern Recognition*, 105:107369, 2020. 2

Schölkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., and Mooij, J. M. On causal and anticausal learning. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012. URL `http://icml.cc/2012/papers/625.pdf`. 1

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. Towards causal representation learning. *CoRR*, abs/2102.11107, 2021. URL `https://arxiv.org/abs/2102.11107`. 1

Schott, L., von Kügelgen, J., Träuble, F., Gehler, P., Russell, C., Bethge, M., Schölkopf, B., Locatello, F., and Brendel, W. Visual representation learning does not generalize strongly within the same domain. In *ICLR 2021 - Workshop on Generalization beyond the training distribution in brains and machines*, May 2021. URL `https://raw.githubusercontent.com/iclr2021generalization/iclr2021generalization.github.io/main/assets/pdf/papers/02.pdf`. 1, 2

Schwarz, K., Liao, Y., Niemeyer, M., and Geiger, A. GRAF: generative radiance fields for 3d-aware image synthesis. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/e92e1b476bb5262d793fd40931e0ed53-Abstract.html`. 2

Shahtalebi, S., Gagnon-Audet, J.-C., Laleh, T., Faramarzi, M., Ahuja, K., and Rish, I. Sand-mask: An enhanced gradient masking strategy for the discovery of invariances in domain generalization. *arXiv preprint arXiv:2106.02266*, 2021. 2

Shi, Y., Seely, J., Torr, P. H. S., Siddharth, N., Hannun, A., Usunier, N., and Synnaeve, G. Gradient matching for domain generalization. *arXiv preprint arXiv:2104.09937*, 2021. 2

Sitzmann, V., Zollhöfer, M., and Wetzstein, G. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 1119–1130, 2019. URL `https://proceedings.neurips.cc/paper/2019/hash/b5dc4e5d9b495d0196f61d45b26ef33e-Abstract.html`. 2

Sitzmann, V., Martel, J., Bergman, A., Lindell, D., and Wetzstein, G. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems*, 33, 2020. 2

Stelzner, K., Kersting, K., and Kosiorek, A. R. Decomposing 3d scenes into objects via unsupervised volume segmentation. *CoRR*, abs/2104.01148, 2021. URL `https://arxiv.org/abs/2104.01148`. 1, 2, 19, 22

Tewari, A., Fried, O., Thies, J., Sitzmann, V., Lombardi, S., Sunkavalli, K., Martin-Brualla, R., Simon, T., Saragih, J., Nießner, M., et al. State of the art on neural rendering. In *Computer Graphics Forum*, volume 39, pp. 701–727. Wiley Online Library, 2020. 2

Trevithick, A. and Yang, B. GRF: learning a general radiance field for 3d scene representation and rendering. *CoRR*, abs/2010.04595, 2020. URL `https://arxiv.org/abs/2010.04595`. 2

van Steenkiste, S., Kurach, K., Schmidhuber, J., and Gelly, S. Investigating object compositionality in generative adversarial networks. *Neural Networks*, 130:309–325, 2020. doi: https://doi.org/10.1016/j.neunet.2020.07.007. 2

Vapnik, V. Principles of risk minimization for learning theory. *Advances in neural information processing systems*, 4, 1991. 1

von Kügelgen, J., Ustyuzhaninov, I., Gehler, P. V., Bethge, M., and Schölkopf, B. Towards causal generative scene models via competition of experts. *CoRR*, abs/2004.12906, 2020. URL `https://arxiv.org/abs/2004.12906`. 2, 22

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 681–688. Citeseer, 2011. 4

Wiles, O., Gowal, S., Stimberg, F., Rebuffi, S.-A., Ktena, I., Dvijotham, K. D., and Cemgil, A. T. A fine-grained analysis on distribution shift. In *ICLR*, 2022. 2

Wizadwongsa, S., Phongthawee, P., Yenphraphai, J., and Suwajanakorn, S. Nex: Real-time view synthesis with neural basis expansion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

Xue, Y., Li, Y., Singh, K. K., and Lee, Y. J. Giraffe hd: A high-resolution 3d-aware generative model, 2022. URL `https://arxiv.org/abs/2203.14954`. 3

Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Basri, R., and Lipman, Y. Multiview neural surface reconstruction by disentangling geometry and appearance. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/1a77befc3b608d6ed363567685f70e1e-Abstract.html`. 2

Ye, N., Li, K., Hong, L., Bai, H., Chen, Y., Zhou, F., and Li, Z. Ood-bench: Benchmarking and understanding out-of-distribution generalization datasets and algorithms. *CoRR*, abs/2106.03721, 2021. URL `https://arxiv.org/abs/2106.03721`. 2

Yen-Chen, L. Nerf-pytorch. `https://github.com/yenchenlin/nerf-pytorch/`, 2020. 19

Yu, A., Ye, V., Tancik, M., and Kanazawa, A. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4578–4587, 2021. 2

Yu, H.-X., Guibas, L., and Wu, J. Unsupervised discovery of object radiance fields. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=rwE8SshAlxw`. 2, 3, 5, 12, 19, 22

Yuille, A. L. and Liu, C. Deep nets: What have they ever done for vision? *International Journal of Computer Vision*, 129(3):781–802, 2021. 2

Supplementary Material

6. Additional Results

In Tab. 2, we report quantitative results for each out-of-distribution (OOD) dataset split. We report all metrics as described in the main paper (Sec. 4), also including standard deviations. It shows that our framework significantly outperforms the baselines on most metrics. In cases of a worse performance, these are mostly within one standard deviation, hence not statistically significant.

We also show qualitative results in Fig. 4 to visually inspect the performance of all models on GQN dataset.

Particularly noteworthy is the significant improvement in the challenging OOD viewpoint setting – a dataset containing images taken from a radically different viewpoint than those seen during training. As noted in prior works (Alcorn et al., 2019; Hinton, 2013; 2021; Kosiorek et al., 2019), such OOD images produce very different representations in a standard neural network; this does not apply to our framework which does not use an encoder network (and hence amortised inference) but instead performs inference directly over the causal generative model. In Tab. 2, we see that on OOD viewpoint split our model achieves higher PSNR and segmentation scores than all other baselines. In the Fig. 4 (rows 7 and 8), we see that both uORF* (Yu et al., 2022) and Slot-Attention (Locatello et al., 2020) fail to reconstruct the input image.

Our model also achieves better PSNR score than all baselines on the OOD composition dataset. Fig. 4 illustrates that baselines (last two columns) again fail the relatively straightforward task of reconstructing the input image. Note that uORF* continues to predict scenes that are similar to those in the training set, despite the inputs being significantly different. For example, in the first two rows, uORF* incorrectly outputs objects with a colour that were present with an orange background in the training set. This is in line with other reports in the literature (e.g. (Geirhos et al., 2020)) showing that potentially the baseline has learnt a ‘shortcut’, to preemptively output a red object when the background pixels are orange.

Similarly, uORF* predicts incorrect positions of objects in the OOD position split (rows 5 and 6) in Fig. 4– in fact, predicting them on the opposite side of the room (as they appear in the training set). In contrast, our model successfully localises objects – we attribute this to our inference mechanism and causal model which represents object with appearance disentangled from position.

In Fig. 5–7, we show generation results for our proposed method and for NeRF-VAE*. These are examples of scenes sampled *a priori*, rendered from multiple viewpoints. We see that in accordance with the quantitative results in the main text, our model is able to sample significantly more realistic scenes (and thus images) than NeRF-VAE*.

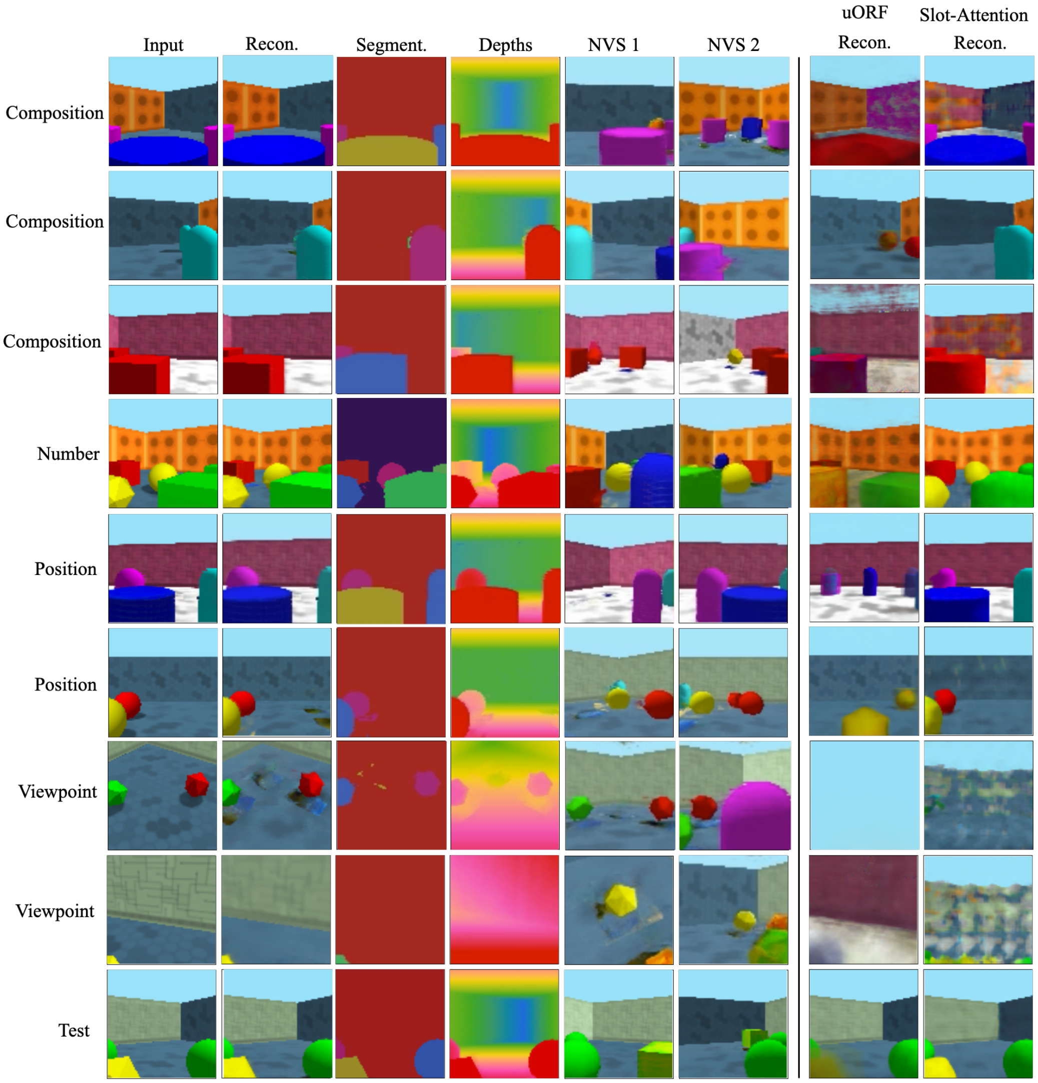


Figure 4. Additional qualitative results from our method and baselines (uORF* and Slot Attention). See the main paper for a discussion of tasks and metrics, and supplementary Sec. 6 for discussion of successes and failures.

	GQN						**ARROW**					
	per-image				per-scene		per-image				per-scene	
	PSNR ↑	D.MRE ↓	ARI ↑	mSC ↑	PSNR ↑	D.MRE ↓	PSNR ↑	D.MRE ↓	ARI ↑	mSC ↑	PSNR ↑	D.MRE ↓
composition												
Ours	**23.3 ± 2.9**	**0.03 ± 0.003**	0.82 ± 0.22	**0.9 ± 0.07**	**19.6 ± 1.9**	**0.067 ± 0.047**	**26.5 ± 0.9**	0.09 ± 0.018	**0.73 ± 0.06**	**0.82 ± 0.02**	**26.1 ± 0.9**	0.09 ± 0.007
Slot Att.	23.1 ± 2.2	–	**0.88 ± 0.2**	0.63 ± 0.11	–	–	25 ± 2.3	–	0.27 ± 0.14	0.13 ± 0.03	–	–
uORF*	13.6 ± 3	0.18 ± 0.158	0.43 ± 0.3	0.43 ± 0.15	13.3 ± 2	0.208 ± 0.105	25.3 ± 1.7	**0.083 ± 0.019**	0.57 ± 0.14	0.39 ± 0.12	25.4 ± 1.4	**0.083 ± 0.006**
NeRF-VAE*	15.1 ± 3.2	0.074 ± 0.101	–	–	14.5 ± 1.5	0.114 ± 0.069	19.2 ± 1.6	0.993 ± 0	–	–	19.2 ± 1.2	0.993 ± 0.0003
position												
Ours	21.7 ± 2.5	**0.033 ± 0.008**	**0.72 ± 0.3**	**0.86 ± 0.09**	**19.2 ± 2.2**	**0.083 ± 0.079**	**26.5 ± 0.6**	0.086 ± 0.015	**0.69 ± 0.05**	**0.8 ± 0.03**	**26.3 ± 0.5**	0.085 ± 0.006
Slot Att.	**23.4 ± 3.1**	–	0.67 ± 0.34	0.63 ± 0.14	–	–	21.4 ± 0.9	–	0.36 ± 0.17	0.14 ± 0.03	–	–
uORF*	12.7 ± 2.7	0.281 ± 0.226	0.14 ± 0.18	0.26 ± 0.12	12.5 ± 2.2	0.34 ± 0.132	20.3 ± 1	**0.076 ± 0.016**	0.32 ± 0.17	0.28 ± 0.05	20.4 ± 0.8	**0.076 ± 0.003**
NeRF-VAE*	12.7 ± 2.7	0.267 ± 0.225	–	–	12.6 ± 2.2	0.329 ± 0.136	19.5 ± 1.2	0.993 ± 0	–	–	19.5 ± 0.8	0.993 ± 0.0003
number												
Ours	23.1 ± 3	**0.034 ± 0.01**	0.67 ± 0.19	**0.87 ± 0.08**	**19.4 ± 2.4**	**0.063 ± 0.031**	**27.1 ± 0.8**	0.087 ± 0.017	**0.46 ± 0.04**	**0.83 ± 0.02**	**26.8 ± 0.5**	0.087 ± 0.006
Slot Att.	**24.0 ± 2.4**	–	**0.74 ± 0.27**	0.6 ± 0.11	–	–	23.8 ± 1.5	–	0.27 ± 0.13	0.16 ± 0.03	–	–
uORF*	18.4 ± 3.2	0.128 ± 0.097	0.49 ± 0.33	0.44 ± 0.15	17.3 ± 2.1	0.14 ± 0.045	22.7 ± 1.3	**0.081 ± 0.018**	0.29 ± 0.21	0.45 ± 0.09	22.7 ± 1	**0.081 ± 0.004**
NeRF-VAE*	20.5 ± 3.9	0.094 ± 0.067	–	–	18.7 ± 2.5	0.114 ± 0.032	19.5 ± 1.5	0.993 ± 0	–	–	19.6 ± 0.9	0.993 ± 0.0003
viewpoint												
Ours	**19.0 ± 4.4**	**0.035 ± 0.051**	**0.61 ± 0.44**	**0.91 ± 0.1**	**15.5 ± 2.5**	**0.072 ± 0.059**	**25.6 ± 3.1**	0.395 ± 0.701	**0.64 ± 0.12**	**0.77 ± 0.08**	**24.4 ± 1**	0.389 ± 0.007
Slot Att.	11.7 ± 1.7	–	0.42 ± 0.44	0.45 ± 0.15	–	–	19 ± 1.8	–	0.16 ± 0.09	0.11 ± 0.02	–	–
uORF*	10.7 ± 3.6	0.581 ± 0.301	0.57 ± 0.48	0.58 ± 0.27	10.5 ± 2.6	0.593 ± 0.195	22.5 ± 3.3	**0.39 ± 0.679**	0.52 ± 0.19	0.4 ± 0.1	22.2 ± 1.1	**0.382 ± 0.009**
NeRF-VAE*	10.8 ± 3.4	0.636 ± 0.246	–	–	10.5 ± 2.1	0.662 ± 0.058	19.3 ± 1.6	0.985 ± 0.008	–	–	18.9 ± 0.6	0.985 ± 0.0001

Table 2. Quantitative results on discriminative tasks, comparing performance for different methods on OOD data splits. Dashes indicate the method does not support the task.

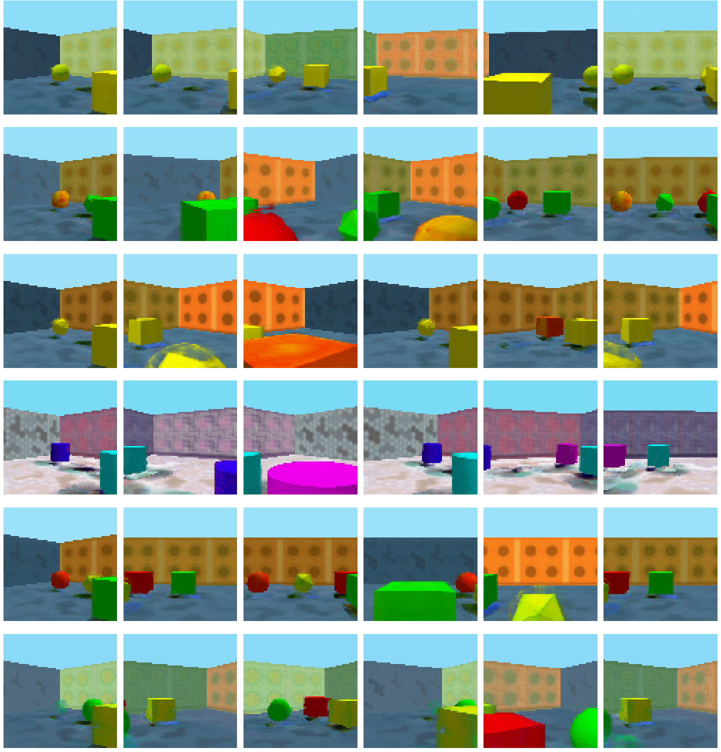


Figure 5. Additional generation results from our method on GQN. Each row corresponds to a single scene, with each column showing a different viewpoint

7. Training Details

We train our generative model from a dataset of images containing K views for each of T scenes. The model includes three learnable components, with parameters θ: (i) $f^*_\theta(\mathbf{q};\, \mathbf{z}^{shape}, \mathbf{z}^{col})$ that represents a 3D object as a function from position to color and density conditioned on the object appearance embedding; (ii) $f^{bg}_\theta(\mathbf{q};\, \mathbf{z}^{shape}_{bg}, \mathbf{z}^{color}_{bg})$ that similarly represents the 3D background; (iii) ζ_θ and ξ_θ, that map the global scene latent $\mathbf{z}^g$ to parameters of the object and background latents $\mathbf{z}^s$. We train these components using autoencoding variational Bayes (Kingma & Welling, 2014; Rezende et al., 2014). The posteriors over Gaussian latent variables are all diagonal Gaussians (parametrized by mean and log-variance), whilst for positions the posterior is Gumbel-Softmax (parametrized by logits). We use two encoder networks to parametrize these variational posteriors. $\mathrm{enc}^s_\phi(\{\mathbf{x}_n, \mathbf{v}_n\}_{n=1}^M)$ parametrizes $q(\mathbf{z}^s|\{\mathbf{x}_n, \mathbf{v}_n\}_{n=1}^M)$; for efficiency, we pass it only a subset of $M < K$ images. It encodes each observed image and its viewpoint $(\mathbf{x}_n, \mathbf{v}_n)$ independently then sums the results (as in (Eslami et al., 2018)) before outputting the posterior parameters; this ensures the encoder is invariant to the ordering of images. $\mathrm{enc}^g_\phi(\mathbf{z}^s)$ parametrizes $q(\mathbf{z}^g|\mathbf{z}^s)$, and takes the lower-level latent code $\mathbf{z}^s$ as input.

For stable training, we adopt a two-stage approach. We first train the model to reconstruct $\mathbf{x}_{1...K}$, via the object-level latent space $\mathbf{z}^s$, ignoring the scene-level latent $\mathbf{z}^g$ and placing standard Gaussian priors on $\mathbf{z}^s$ (c.f. (Jiang & Ahn, 2020)), i.e. maximizing the following evidence lower-bound (ELBO):

$$\mathcal{L}^s = \mathbb{E}_{q_\phi(\mathbf{z}^s|\{\mathbf{x}_n,\mathbf{v}_n\}_{n=1}^M)}\left[\sum_{n=1}^{K} \log f_{\mathcal{N}}(\mathbf{x}_n;\, C(S_\theta(\mathbf{z}^s),\, \mathbf{v}_n),\, \sigma^2)\right] - D_{\mathrm{KL}}\left[q_\phi(\mathbf{z}^s \mid \{\mathbf{x}_n,\mathbf{v}_n\}_{n=1}^M) \,\|\, \mathcal{N}(\mathbf{0},\, \mathbf{1})\right] \tag{5}$$

After this has converged, we learn the scene-level latent space by maximizing

$$\mathcal{L}^g = \mathbb{E}_{q_\phi(\mathbf{z}^g|\mathbf{z}^s)}\left[\mathbb{E}_{q_\phi(\mathbf{z}^s|\{\mathbf{x}_n,\mathbf{v}_n\}_{n=1}^M)} \log p_\theta(\mathbf{z}^s \mid \mathbf{z}^g)\right] - D_{\mathrm{KL}}\left[q_\phi(\mathbf{z}^g \mid \mathbf{z}^s) \,\|\, p_\theta(\mathbf{z}^g)\right] \tag{6}$$

We use Adam for optimization (Kingma & Ba, 2015), β-weighting of KL terms (Higgins et al., 2017), and approximate each of the above expectations by a single sample. We also further approximate $\mathcal{L}^s$ by rendering only a random subset of pixels per minibatch.

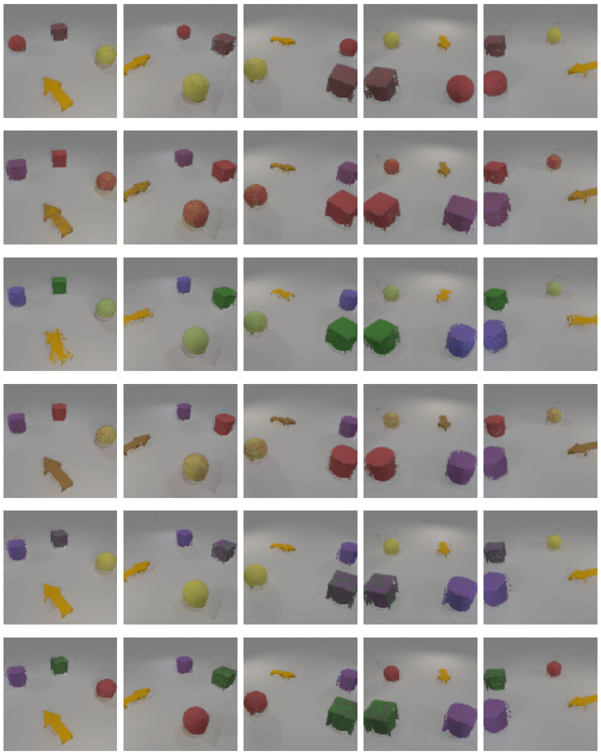


Figure 6. Additional generation results from our method on ARROW. Each row corresponds to a single scene, with each column showing a different viewpoint

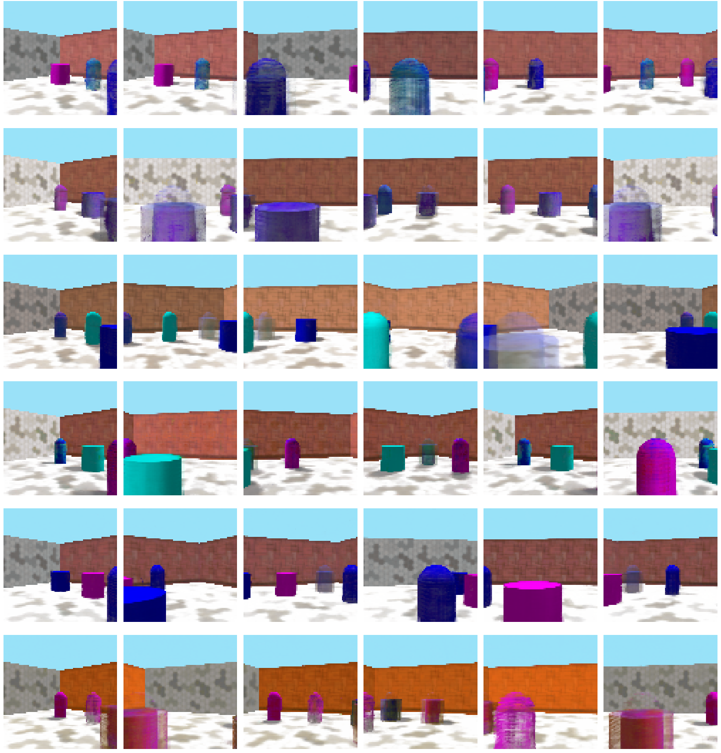


Figure 7. Generation results from NeRF-VAE* on GQN. Each row corresponds to a single scene, with each column showing a different viewpoint

8. Architecture Details

The variational posterior $q_\phi(\mathbf{z}^s \mid \{\mathbf{x}_n, \mathbf{v}_n\}_{n=1}^M)$ used during training is parameterised by an image encoder, pose encoder and pooled representation encoder. For the image encoder, we use a convolutional neural network:

Layer	Filters	Stride	Norm./Act.
Conv 4×4	16	1	Layer/CELU
Conv 3×3	16	1	Layer/CELU
Conv 4×4	32	2	Layer/CELU
Conv 3×3	32	1	Layer/CELU
Conv 4×4	64	2	Layer/CELU
Conv 3×3	64	1	Layer/CELU
Conv 4×4	128	2	Layer/CELU
Conv 3×3	128	1	Layer/CELU
Conv 4×4	128	2	Layer/CELU

For the camera pose encoder, we use a fully-connected neural network:

Layer	Size	Norm./Act.
MLP	50	Layer/CELU
MLP	50	Layer/CELU
MLP	50	Layer/CELU
MLP	50	Layer/CELU
MLP	50	Layer/CELU
MLP	50	–

We then pool the image and camera representations – we sum-pool $M = 5$ images with their corresponding viewpoints $\{\mathbf{x}_n, \mathbf{v}_n\}_{n=1}^M$ for GQN dataset and use $M = 1$ for Arrow dataset. The pooled representation is taken as input to the scene representation encoder, which outputs shape and appearance embeddings for each component, and an embedding k for each object's position:

Layer	Size	Norm./Act.
MLP	116	Layer/CELU
MLP	116	Layer/CELU
MLP	116	Layer/CELU

The embedding k is then passed through another fully-connected neural network to output logits for a Gumbel Softmax (relaxed categorical variable) one-hot position indicator.

For GQN and Arrow datasets, object shape and appearance embeddings are both 9 dimensional. Background appearance is 3 dimensional for GQN whilst we use 1 dimension for Arrow. Object appearance and shape embeddings are variationally autoencoded to a $10-$dimensional object Gaussian variable while background appearance and shape embeddings are variationally autoencoded to a 1-dimensional Gaussian variable. Encoders and decoders for both are residual neural networks with the following layers:

Layer	Size	Norm./Act.	Residual Connection
MLP	150	Layer/CELU	No
MLP	150	Layer/CELU	Yes
MLP	150	Layer/CELU	Yes
MLP	150	Layer/CELU	Yes
MLP	150	Layer/CELU	Yes
MLP	150	Layer/CELU	Yes
MLP	150	Layer/CELU	Yes

We similarly variationally autoencode object-level scene representation to a global scene variable. $\mathbf{z}^g$ is a Gaussian variable with 10 dimensions for Arrow and 20 dimensions for GQN. Encoder and decoder are fully-connected residual neural networks with the following layers:

For Arrow:

Layer	Size	Norm./Act.	Residual Connection
MLP	300	Layer/CELU	No
MLP	300	Layer/CELU	Yes
MLP	300	Layer/CELU	Yes
MLP	300	Layer/CELU	Yes
MLP	300	Layer/CELU	Yes
MLP	300	Layer/CELU	Yes
MLP	300	Layer/CELU	Yes

For GQN:

Layer	Size	Norm./Act.	Residual Connection
MLP	400	Layer/CELU	No
MLP	400	Layer/CELU	Yes
MLP	400	Layer/CELU	Yes
MLP	400	Layer/CELU	Yes
MLP	400	Layer/CELU	Yes
MLP	400	Layer/CELU	Yes
MLP	400	Layer/CELU	Yes
MLP	400	Layer/CELU	Yes
MLP	400	Layer/CELU	Yes
MLP	400	Layer/CELU	Yes

The 3D shape and appearance of the object and background components are represented as Neural Radiance Fields (NeRFs) (Mildenhall et al., 2022), conditioned on the component's appearance and shape embeddings. We define two NeRF MLPs – one for background component and one for object component. As shown in the following figure Fig. 8, to facilitate disentanglement of shape and appearance, the opacity σ depends only on shape embedding, while the radiance $\mathbf{c}$ depends only on appearance embedding.

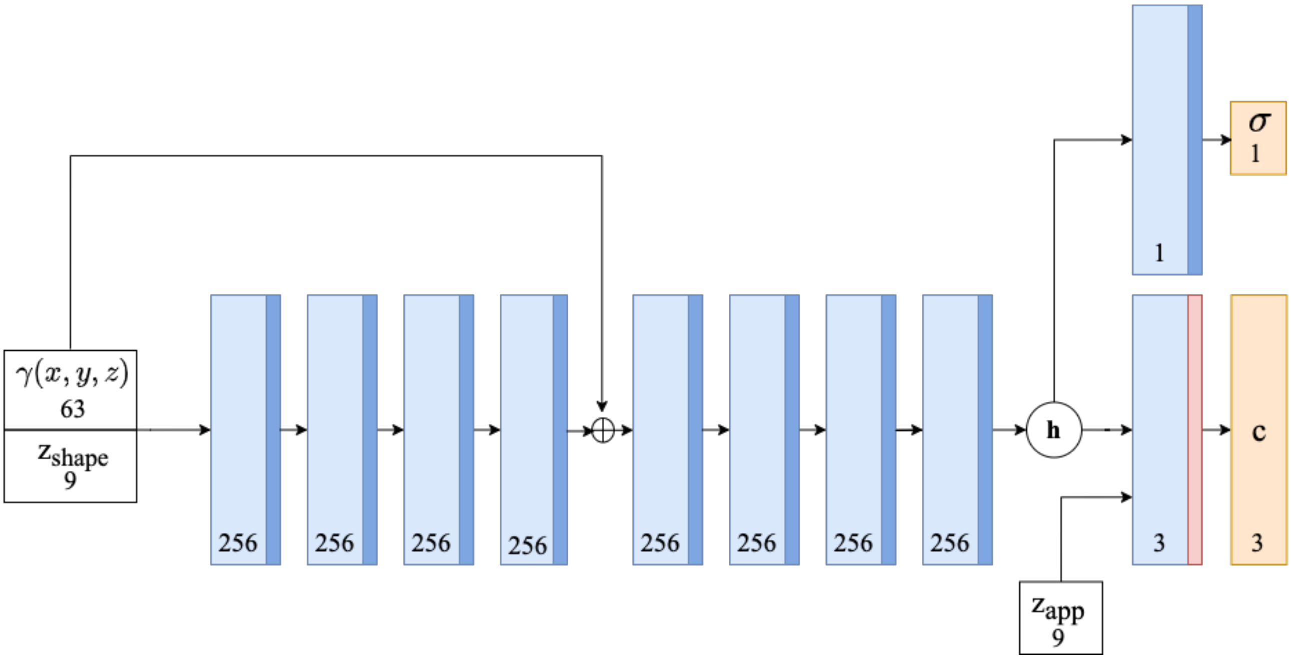


Figure 8. Architecture of a neural radiance field (Mildenhall et al., 2022; Niemeyer & Geiger, 2021) conditioned on component's shape and appearance codes.

Aside from these conditioning vectors, we use the vanilla NeRF architecture used in (Mildenhall et al., 2022). 3D points are passed through a standard positional embedding which outputs a 63-dimensional embedding. Then the first part of the network takes a positional embedding of 3D point $\gamma(x, y, z)$ and a shape embedding and passes it through 8 fully-connected layers to output a feature vector h:

Layer	Size	Act.
MLP	256	ReLU
MLP	256	ReLU
MLP	256	ReLU
MLP	256	ReLU
MLP	256	ReLU
MLP	256	ReLU
MLP	256	ReLU
MLP	256	ReLU

Then h is passed through a linear layer with ReLU activation to output the density σ. Then h concatenated with appearance embedding is passed through a linear layer with sigmoid activation to output the radiance **c**. We initialise the network with a bias of 3.0 before the output of density σ. During training we add Gaussian noise with standard deviation of 0.3 to the output of σ.

9. Implementation Details

Our final models were trained on a single NVIDIA A100 GPU for approximately 4 weeks (we did not measure the total compute time for development). We implemented our model with PyTorch (Paszke et al., 2019). We train our model with Adam (Kingma & Ba, 2015) using a learning rate of $1\mathrm{e}{-4}$. During the first stage of training, we use a batch size of 8. During training, we subsample and render 8000 pixels, and for each pixel, we sample 512 3D points along a ray (we use 256 points initially to speed up training) from a distance of 0.01 to 20.0 and 9.9 for Arrow and GQN respectively. Most other rendering hyperparameters are identical to those of (Mildenhall et al., 2022; Yen-Chen, 2020). During a second stage of training, we use batch size of 64. We multiply the KL-divergence term in ELBO loss by β (Higgins et al., 2017): (i) for scene-level variable we use $\beta = 50$ for GQN and $\beta = 10$ for Arrow (ii) for object-level variable we use $\beta = 5$ for both Arrow and GQN.

We performed hyperparameter optimization with a grid search over the learning rate $(1\mathrm{e}{-5}, 3\mathrm{e}{-5}, 5\mathrm{e}{-5}, 7\mathrm{e}{-5}, 9\mathrm{e}{-5}, 1\mathrm{e}{-4}, 3\mathrm{e}{-4}, 5\mathrm{e}{-4}, 7\mathrm{e}{-4}, 9\mathrm{e}{-4}, 1\mathrm{e}{-3}, 3\mathrm{e}{-3}, 5\mathrm{e}{-3}, 7\mathrm{e}{-3}, 9\mathrm{e}{-3})$, batch size $(1, 2, 4, 8, 16, 32, 64, 128)$, number of subsampled/rendered pixels used for estimating the loss $(4000, 8000, 12000, 16000, 20000)$, number of sampled 3D points along a ray $(64, 128, 256, 512)$.

During training, foreground is modelled with 5 object components and one component is used for modelling the background. During test-time, the number of object components is fixed to the maximum number of objects present in any dataset split. During both train and test time, we provide the model with possible object candidate positions (Henderson & Lampert, 2020; Chen et al., 2021) and fix the background component to be outside the possible foreground range in a dataset (Henderson & Lampert, 2020; Yu et al., 2022; Stelzner et al., 2021). We have initially performed experiments without such inductive biases, finding that model performs well only in around 1 in 10 runs; with the rest of runs model captures background with foreground components and vice-versa.

Rendering a scene during training assumed that a random light is being emitted from the world (in contrast to assumption of black or white world in (Mildenhall et al., 2022)) – this prevents model from exploiting the world's emitted colour to model the scene and encourages modelling non-trivial scene to avoid random light from entering the camera.

The Gaussian mixture model is trained by Expectation Maximisation (Dempster et al., 1977). Each component has its own general covariance matrix. For GQN, we use 20 components for background variable and 20 components for the object variable. For Arrow, we use 5 components for background variable and 70 components for the object variable

The image is modelled with a Normal distribution with a mean outputted by a model as described in the method Section Sec. 3.1 and a standard deviation of 0.15. During MCMC inference, the gradient descent step has a learning rate of $1\mathrm{e}{-4}$. The variatiotional autoencoders model their output with a Gaussian distribution with a fixed standard deviation of 0.01.

10. Datasets

We now describe the datasets that we used for training and evaluation.

10.1. GQN

We render images of shape $(80, 80, 3)$ of rooms containing several objects (icosahedrons, cubes, capsules, cylinders, spheres), based on the 'rooms ring camera' dataset of (Eslami et al., 2018); similar datasets were used in (Henderson & Lampert, 2020; Engelcke et al., 2020), but in all cases without OOD test splits. For evaluation with each OOD dataset split, we sample one image from each scene and use the sampled image for per-image metrics while using all 30 for per-scene metrics. The data generation source code will be made public.

Training split. The training split contains 10000 scenes, each with 30 RGB images. The camera viewpoints are on a circular path around the center of the room with the elevation of 1.0 and with the camera pointed at the center of the room at the same elevation angle. The first camera viewpoint along circular path is generated by first sampling a random initial yaw of the camera with respect to the origin of the scene and then shifting a camera in xy (horizontal) plane by a random distance (sampled uniformly from $\mathcal{U}_{[3.1,3.4]}$) from the origin. Subsequent viewpoints form a circular path with respect to the origin. Textures for the walls and colors for the objects are selected randomly from a finite set, with some combinations held out. Three walls have the same texture as each other, with the fourth different. In particular, the training split contains scenes with either (i) random compositions of (light, cerise) background textures with (capsule, cylinder) objects with 3 random colours or (ii) random compositions of (cosahedron, box, sphere) objects with another 3 random colours and with (orange, blue, green) background textures. There are 3–4 objects present; these are placed near the side of the room identified by the odd background texture.

OOD composition split. The OOD composition split contains 100 scenes, each with 30 RGB images. The data generation process is same as for training split, except that we swap possible background textures. Hence, the split contains scenes with either (i) random compositions of (orange, blue, green) background textures with (capsule, cylinder) objects with 3 random colours or (ii) random compositions of (cosahedron, box, sphere) objects with another 3 random colours with (light, cerise) background textures.

OOD position split. The OOD position split contains 100 scenes, each with 30 RGB images. The data generation process is same as for training split, except that objects are now placed in the opposite side of the room identified with the odd background texture.

OOD viewpoint split. The data generation process is same as for training split, except that camera viewpoint is sampled from a different procedure. In particular, we have two OOD viewpoint splits, each containing 100 scenes with 30 RGB images. The first part contains camera elevation sampled from uniform distribution $\mathcal{U}_{[0.1,4.0]}$ and position sampled randomly in between objects. The camera has a random yaw and its pitch is such that the camera is focused on the point with elevation of 1.0 at a distance of 1.0. A second part of OOD viewpoint split contains images with camera pointing from high-up to the center of the room with a circular path around the origin. For each frame, the camera has a xy (horizontal) distance of 1.5 from the origin, elevation of 2.5 and a pitch of -0.75.

OOD number split. The OOD number split contains 300 scenes, each with 30 RGB images. The data generation process is same as for training split, except that we now have 1, 5 or 6 objects (100 scenes per each) instead of 3-4 objects. Objects are placed randomly in the opposite side of the room identified with the odd background texture, but in contrast to training split, now some objects can be placed in the middle of the room to avoid cluttering the scene.

10.2. ARROW

We render RGB images of shape $(96, 96, 3)$ using a modified version of the CLEVR dataset (Johnson et al., 2017), similar to those in (Jiang & Ahn, 2020). Similarly to (Kosiorek et al., 2021; Nanbo et al., 2020), we modify background component such that the background is present from all camera viewpoints.

Training split. The training dataset contains 10000 scenes, with 20 images per each scene. Each scene has four objects, of which one is always an arrow, two of which are the same as each other, and a fourth that is different. The arrow always points at the odd-shaped (fourth) object. The colours of objects are randomly sampled. The camera points to the origin $(0, 0, 0)$ from two circular path around it: one starting at position of $(4.99, -4.33, 4.89)$, a second starting at $(3.75, 3.75, 1.0)$.

OOD composition split. The OOD composition dataset contains 100 scenes, with 10 images per each scene. For evaluation, we sample one image per scene and use it to evaluate per-image metric while using all 10 images for per-scene metrics. The data generation process is same as for training split, except that now all objects contain same shape and colour, with no arrow present.

OOD position split. The OOD position dataset contains 100 scenes, with 10 images per each scene. For evaluation, we sample one image per scene and use it to evaluate per-image metric while using all 10 images for per-scene metrics. The data generation process is same as for training split, except that now four objects are positioned in a an approximate line, and the arrow no longer points to the odd object.

OOD viewpoint split. The OOD position dataset contains 100 scenes, with 21 images per each scene. It contains one image rendered from the top at position $(0., 0., 8.0)$ to the origin $(0, 0, 0)$. The other 20 images are from two circular paths around the origin: one starting at position of $(4.75, 4.75, 0.4)$, a second starting at $(0.5, 0.5, 8.0)$. Both starting positions are with added noise from uniform distribution $\mathcal{U}_{[-0.5,0.5]}$ to each dimension.

OOD number split. The OOD position dataset contains 100 scenes, with 10 images per each scene. The data generation process is same as for training split, but now the scene contains 1, 5 or 6 objects instead of 4.

11. Baselines

We used the original publicly-available implementation of (Locatello et al., 2020), with hyperparameters (including number of slots and number of iterations) re-tuned on our datasets. For the 80×80 GQN images, we slightly modified the decoder architecture, increasing the initial feature map size. As for our method, at test time, we set the number of slots equal to the largest number of objects (plus one for background) in any OOD split.

For a fair comparison with NeRF-VAE* and uORF*, we provide these methods with the same inductive biases as we used for our method. In particular, we specify possible scene and foreground boundary and use the same number of samples along a ray as described in Implementation Details section. Furthermore, to make these methods work on the GQN dataset, we have found that it is useful to provide multiple images to the encoder during training. However, during test time, we evaluate all methods on a single image. We hence first pretrain both approaches with multiple views with the same pooling mechanism as described in our method's implementation details. We then interrupt the training and continue training the model with encoder taking $M = 1$ images as input. We also provide baselines with an equivalent architecture to ours.

12. Related Work

In Tab. 3, we enumerate various works addressing similar tasks to our proposed approach, and note whether they support various tasks/features.

Table 3. Comparison of capabilities of related models. Check (✓) - capability supported by the model; cross (✗) - capability not supported by the model.

	AIR (Eslami et al., 2016)	GQN (Eslami et al., 2018)	MONET (Burgess et al., 2019)	IODINE (Greff et al., 2019)	GENESIS (Engelcke et al., 2020)	SCALOR (Jiang et al.)	SPACE (Lin et al., 2020)	OCF (Anciukevičius et al., 2020)	GNM (Jiang & Ahn, 2020)	ROOTS (Chen et al., 2021)	RELATE (Ehrhardt et al., 2020)	MulMON (Nanbo et al., 2020)	ECON (von Kügelgen et al., 2020)	O3V (Henderson & Lampert, 2020)	GENESIS-V2 (Engelcke et al., 2021)	GIRAFFE (Niemeyer & Geiger, 2021)	uORF (Yu et al., 2022)	NeRF-VAE (Kosiorek et al., 2021)	SIMONE (Kabra et al., 2021)	ObSuRF (Stelzner et al., 2021)	*ours*
explicitly represents 3D shapes	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓	✓	✓	✗	✓	✓
explicitly represents 3D object positions	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✓	✗	✓	✗	✗	✗	✗	✓
infers representation given an image	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✗	✓	✗	✓	✓	✓	✓	✓
generates new plausible images/scenes	✗	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✓	✓	✗	✗	✓	✗	✗	✓
learns a prior over individual object appearances	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✗	✗	✗	✗	✓
explicit volumetric rendering	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓	✓	✗	✓	✓