Unsupervised Causal Generative Understanding of Images

Anonymous Author(s)
Affiliation
Address
email

Abstract

We present a novel framework for unsupervised object-centric 3D scene understanding that generalizes robustly to out-of-distribution images. To achieve this, we design a causal generative model reflecting the physical process by which an image is produced, when a camera captures a scene containing multiple objects. 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. It explicitly represents object instances as separate neural radiance fields, placed into a 3D scene. We then propose an inference algorithm that can infer this latent representation given a single out-of-distribution image as input – even when it shows an unseen combination of components, unseen spatial compositions or a radically new viewpoint. We conduct extensive experiments applying our approach to test datasets that have zero probability under the training distribution. These show that it accurately reconstructs a scene's geometry, segments objects and infers their positions, despite not receiving any supervision. Our approach significantly out-performs baselines that do not capture the true causal image generation process.

1 Introduction

Most machine learning approaches – both supervised and unsupervised – make the assumption that at test time, they are applied to data drawn from the same distribution as seen during training [12, 55, 17, 26]. This means the powerful generalization guarantees of statistical learning theory apply [101]. Indeed, 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 [3, 38, 39, 7, 53], shifted by few pixels [6], and showing scenes with an unseen composition of objects [9, 86, 91, 29, 30]. It has been suggested that this is because they learn spurious correlations – or *shortcuts* [29] – to achieve low training loss, but which do not capture the true *causal* relationships that remain universally valid. However, in the real world, we do commonly encounter observations that are outside of the distribution from which we learnt – indeed, there are exponentially many possible combinations of objects, viewpoints and environments, so we cannot hope to gather data showing every one.

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 challenging – yet practically important – setting where at test time, we see images depicting scenes that have zero probability in the training distribution (Fig. 1). We adopt an *unsupervised* approach to learning, thus avoiding the need for expensive manual annotation of object masks, 3D positions, and similar – we require only a dataset of posed multi-view images.

Our approach is to build a generative model (Sec. 3) jointly over image pixels and the 3D world they depict, that can be robustly inverted to infer the latent factors that gave rise to an input image

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

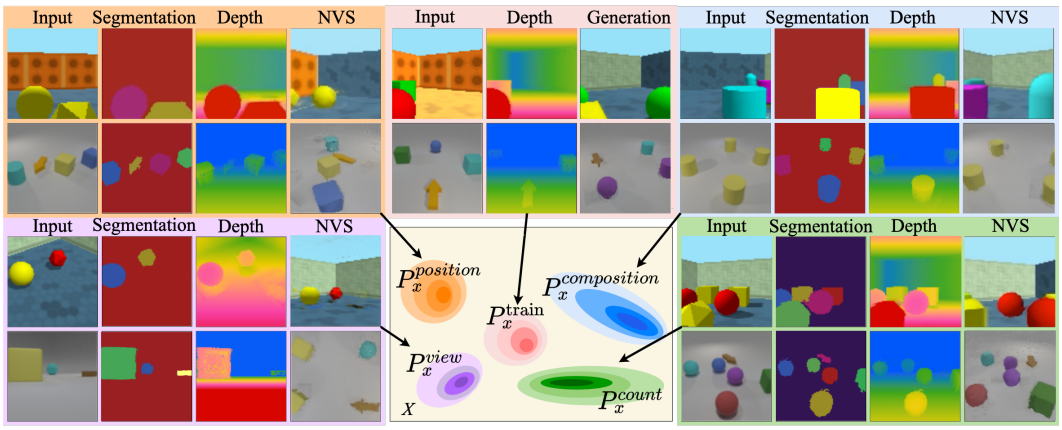

Figure 1: Our generative model is trained to reconstruct multi-view images drawn from its training distribution (red), in terms of a latent object-centric 3D scene representation. After training, it can sample plausible scenes containing multiple objects ('Generation', top center). At test time, it takes as input 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.

[90, 10, 103, 58, 18]. If such a model is to support generalization to out-of-distribution (OOD) data, it should reflect the underlying causal model of the environment from which the data arose [81, 90]. In practice this means its graphical model structure (i.e. conditional independencies) should match those present in the environment, with conditional distributions in the model corresponding (as far as possible) to *independent mechanisms* [89, 44] in the world. In this case, when certain mechanisms or physical processes in the environment change, we still expect much of our model to remain applicable; this has been called *sparse mechanism shift* [90]. For example, even if objects are no longer placed according to the same patterns as at training time, we still expect the appearances of object types to remain consistent – even if chairs are no longer placed near tables, both should still have four legs.

However, in general it is impossible to recover such a causal model purely from observational data [81, 90]. We therefore embed in our model universally-applicable physical knowledge that the world is composed of 3D objects of different shapes, which may appear at different locations, and which are imaged by a camera subject to the laws of 3D geometry and perspective projection. Each object is associated with separate latent variables describing its position and appearance; the use of separate latents ensures disentanglement, so the same object can be represented invariantly in different locations. The final image is synthesized by volumetric rendering of the latent 3D shapes placed in a 3D scene space according to their latent positions. This structure means the model is *compositional*, in the sense that having learnt about one type of object in one context or environment, it can also represent and perform inference about a similar object even in different contexts.

We perform inference on OOD data by *intervening* on our model [81, 83] – i.e. replacing certain conditional distributions (or mechanisms) that are no longer appropriate – then doing posterior inference over the updated model. For example, if object positions no longer follow the distribution seen during training, we replace the learnt prior on locations with a uniform prior (leaving unchanged the conditional distribution on object appearances), then perform inference over this updated model. We must also ensure that the inference method itself supports OOD generalization. This means we cannot employ the common approach of amortized variational inference [50, 85], as an encoder network is in general not robust to changes of distribution [71, 29], and in our setting we do not have a dataset of images from the test-time distribution to learn from. Instead, we develop a novel Markov chain Monte-Carlo (MCMC) inference scheme, that finds posterior samples for a given test image without any encoder network.

Targeting OOD generalization imposes several technical constraints on our model compared with other recent compositional generative models of 3D scenes (e.g. [77, 35, 97]), none of which address this setting. Firstly, we cannot include a learnt neural renderer (mapping features to pixels in image space) [77, 15, 74] in the generative model, as these are not guaranteed to generalise to novel camera viewpoints nor compositions of objects. Similarly, spatial mixture models [26, 73, 97] cannot perform

inference on scenes with objects at unseen positions because their representation of object appearance is entangled with that of position [71, 63].

To evaluate our approach (Sec. 4), we create challenging test datasets that have zero probability under the corresponding training distribution yet share aspects of its structure, such as featuring the same object classes composed in different arrangements. We show that our model can generalize to unseen numbers of objects, unseen compositions, and radically new camera viewpoints – all significantly better than existing works. It successfully localizes objects in the 3D scene, reconstructs their 3D shapes, and hence predicts depth-maps and instance segmentation masks.

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) experimental evidence that current methods do not generalize to images with a radically new camera viewpoint, a different number of objects, or an unseen composition of objects; (ii) a novel generative model of 3D scenes based on multi-object radiance fields with explicit object positions and volumetric rendering; and (iii) 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

Recently numerous works have observed that modern learning-based computer vision methods fail to generalise on OOD data, and have analysed this [38, 3, 39, 53, 6, 9, 86, 91, 29, 30, 72, 69] as well as constructing various benchmarks [58, 7, 32, 48, 109, 105]. Others have tried to improve OOD generalisation, the most relevant by aiming to learn relations that are invariant across datasets [5, 94, 2, 1, 80, 56, 93, 61]. These methods assume that multiple differently-distributed datasets are available during training, and only address the supervised setting (mainly image classification). Like the latter, our focus is on enabling OOD generalisation, but in contrast to them, we address unsupervised object-centric generative modelling. Moreover, we do not rely on multiple *training* datasets to discover causal factors, but instead directly incorporate universal knowledge such as separation of the world into objects.

Our work is also connected to the *vision as inverse graphics* paradigm [31, 8, 87]. 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 [65, 57, 41, 87, 88, 40]. Like our work, these typically use a test-time optimisation; unlike ours, they do not attempt to learn priors on object layout nor shapes – instead assuming these are known *a priori*.

Neural implicit scene representations [98] aim to a learn continuous representation of a 3D scene from 2D images using neural rendering, either by explicit volumetric rendering [70, 67, 106, 108, 78] or with CNN post-processing of rendered features [96, 95]. 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 [110, 99, 59, 28, 82, 76, 43], 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 – meaning manipulating the scene, e.g. moving single objects – is not possible. In contrast, [79] divides a scene into objects, but requires detailed manual annotations to do so, and does not support inference from few images; [22] relies on weaker supervision in the form of ground-truth object masks. [111, 97] discover such a decomposition automatically (though with depth supervision for [97]), however they rely on spatial mixtures to assign each point in space to an object, without any explicit, controllable representation of object positions. This entanglement of latent position and appearance means they are not guaranteed to generalise to OOD combinations of position and appearance [63]. Finally, [33] 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), with no probabilistic model over appearances nor layouts.

Other approaches have extended neural rendering to the generative setting [92, 77, 74, 75, 54, 16, 21, 20], allowing sampling objects or scenes *a priori*. However, these do not allow us to perform inference in the OOD setting, as they have components that do not reflect the causal, compositional structure of the world – they either include a monolithic latent space lacking object-centric representations, or a learnt neural renderer that will not generalise to OOD viewpoints nor compositions. In contrast, there are object-centric generative models that can sample plausible images and perform inference – but

only in 2D, without reasoning over a latent 3D scene representation, and therefore without supporting 3D tasks such as reconstruction depth prediction. Some use a full-image spatial mixture model [26, 73, 51, 24, 25, 47] or alpha stacking [100, 102]; others model images as composed of smaller patches or sprites [27, 45, 4], learning both the distribution of appearances and compositions/layouts; others use compositional energy-based models [23, 62]. A 3D extension of these latter is proposed by [35], using a voxel representation, but this requires videos for inference at test time, and does not support OOD generalisation. In the supplementary, we provide a table comparing the capabilities of our method to closely related works.

3 Method

Our goal is to infer an explicit object-centric representation of a 3D scene from one or multiple images, even when they show scenes lying outside the distribution observed during training. We achieve this by learning a compositional generative causal model jointly over multi-view images and the scenes they depict, as described in Sec. 3.1. For training (Sec. 3.2) we use only posed multi-view images $(\mathbf{x}_1, \mathbf{v}_1), ..., (\mathbf{x}_N, \mathbf{v}_N)$ drawn from a training distribution $P_X^{train}(\mathbf{x}, \mathbf{v})$ without any annotations such as depth-maps, bounding-boxes or segmentation masks. At test time, input images are drawn from a distribution $P_X^{OOD}(\mathbf{x}, \mathbf{v})$ with support disjoint from the training distribution. Performing inference over the generative model using our proposed framework (Sec. 3.3) yields an explicit object-centric representation, including 3D object shapes, positions and appearances.

3.1 Compositional Generative Causal Model

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

Scene representation. The scene $\mathcal{S} = \Big(\mathbf{s}_{bg},\, \{(\mathbf{s}_i^{app},\, \mathbf{s}_i^{pos})\}_{i=1}^{O}\Big)$ is composed of a 3D background component $\mathbf{s}_{bg}$ describing the background's shape and color, and 3D objects indexed $i = 1 \dots O$ 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) [70], 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 most prior work that parametrized 3D object positions as coordinates, we represent them by 1-hot vectors choosing from a set of plausible candidate locations to use as the center position of the object. As we explain in Sec. 3.3, this makes gradient-based optimization easier.

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 [45, 4]. The individual object appearances are specified by Gaussian latent variables $\mathbf{z}_i^{shape}$ and $\mathbf{z}_i^{col}$ that respectively encode the shape and color of the ith object; they are conditioned on $\mathbf{z}^g$, with mean and log-variance given by a fully-connected network $\zeta_\theta(\mathbf{z}^g)$ with weights θ.[1] 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 $\mathbf{z}^s = \{\mathbf{z}_{bg}^{shape},\, \mathbf{z}_{bg}^{col},\, \mathbf{z}_{1\dots O}^{shape},\, \mathbf{z}_{1\dots O}^{col},\, \mathbf{z}_{1\dots O}^{pos}\}$.

The latents $\mathbf{z}^s$ are mapped to the scene $\mathcal{S}$ by a function S_θ. This sets $\mathbf{s}_i^{pos}$ equal to $\mathbf{z}_i^{pos}$, and derives the 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 an image $\mathbf{x}_n$ given its camera viewpoint $\mathbf{v}_n$ is then

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

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

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

[1] See the supplementary for all network architectures

Rendering the scene $\mathcal{S}$. The rendering process $C(\mathcal{S}, \mathbf{v})$ outputs an image $\mathbf{x}$ for a camera viewpoint $\mathbf{v}$, given our explicit compositional representation of a scene $\mathcal{S}$. Recall $\mathcal{S}$ contains a 3D background component $\mathbf{s}_{bg}$ and a set of object components $\{\mathbf{s}_i^{app}, \mathbf{s}_i^{pos}\}_{i=1}^{O}$; for brevity we will identify the background as component $i = 0$, with $\mathbf{s}_0^{pos}$ fixed to the origin. We extend multi-component neural radiance fields (NeRF) [70, 67] to support explicit placement of objects in the 3D scene according to the position variables $\mathbf{s}_i^{pos}$. Specifically, the latent codes $\mathbf{z}_i^{shape}$ and $\mathbf{z}_i^{col}$ for the ith 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:

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

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 [111], 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 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 [67, 68]:

$$C(\mathcal{S},\, \mathbf{v})[\mathbf{r}] = \int_0^\infty T(t) \sum_{i=0}^{O} \sigma_i(\mathbf{r}(t)) \cdot \mathbf{c}_i(\mathbf{r}(t)) \mathrm{d}t\,, \quad \text{where} \quad T(t) = \exp\left(-\int_0^t \sum_{i=0}^{O} \sigma_i(\mathbf{r}(t'))dt'\right) \tag{4}$$

Continuous relaxation of object placement. To allow gradient-based training and inference, our generative process must be differentiable. We therefore relax the categorical position variable to a Gumbel-Softmax [42, 66]. The object placement operation Eq. 3 can then be efficiently implemented as a convolution operation in the Fourier domain – exploiting the Fourier transform's *sifting property* [14, 4]. 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 [77, 107, 60], which can get stuck in local minima if the model has a poor initial prediction, as the gradient of pixels wrt position is zero if the predicted and true positions do not overlap.

3.2 Training

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) $f_\theta^{bg}(\mathbf{q};\, \mathbf{z}_{bg}^{shape}, \mathbf{z}_{bg}^{color})$ 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 [50, 85]. 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. $\mathrm{enc}_\phi^s(\{\mathbf{x}_n, \mathbf{v}_n\}_{n=1}^M)$ 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 [28]) before outputting the posterior parameters; this ensures the encoder is invariant to the ordering of images. $\mathrm{enc}_\phi^g(\mathbf{z}^s)$ 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 $\mathbf{x}_{1\ldots K}$, 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. [45]), i.e. maximizing the following evidence lower-bound (ELBO):

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

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

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

	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 ↓
Test												
Ours	24.1	0.031	0.81	**0.88**	20.8	0.058	27.1	**0.100**	**0.71**	**0.82**	26.8	**0.107**
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**	25.3	0.991	–	–	25.3	0.991
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 discriminative 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**.

We use Adam for optimization [49], β-weighting of KL terms [37], 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. More implementation details are in the supplementary material.

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 [81], 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 [52].[2]

Our novel MCMC scheme alternates Langevin dynamics (LD) [11, 104] and Metropolis-Hastings (MH) [34] 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

$$\log f_{\mathcal{N}}(\mathbf{x}^*;\, C(S_\theta(\mathbf{z}^s),\, \mathbf{v}^*),\, \sigma^2) + \log p_\theta(\mathbf{z}^s \mid \mathbf{z}^g) + \log p(\mathbf{z}^g) \propto \log p(\mathbf{z}^g,\, \mathbf{z}^s \mid \mathbf{x}^*,\, \mathbf{v}^*) \qquad (7)$$

Each MH step first picks an object slot i uniformly at random. It then samples a new latents for that object from a proposal distribution $\tilde{p}(\mathbf{z}_i^{shape},\, \mathbf{z}_i^{col})$, and decides whether to accept this transition according to the usual Metropolis-Hastings acceptance criterion [34]. The proposal distribution $\tilde{p}$ approximates $\frac{1}{J}\sum_{i=1}^{J} p(\mathbf{z}_i^{shape},\, \mathbf{z}_i^{col} \mid \mathbf{z}_g)$ using a Gaussian mixture model fitted by expectation-maximisation [19]; 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. In practice 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 single sample from each chain that had maximum probability.

4 Experiments

We conduct experiments on two synthetic datasets, using our model and three baselines. We first evaluate performance in the standard setting where the test-set distribution matches the training

[2] We also conducted early experiments with black-box variational inference [84], but this performed poorly due to becoming trapped in local minima. MCMC is more able to explore diverse modes of the posterior

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. We first describe these datasets and test splits (Sec. 4.1), the tasks we address and metrics used (Sec. 4.2), baselines we compare against (Sec. 4.3), and finally the performance of our proposed model and the baselines (Sec. 4.4). Implementation details for our model and baselines (hyperparameters, hardware, etc.) are given in the supplementary material.

4.1 Datasets

GQN. We render images of rooms containing several objects (cubes, cylinders, spheres), based on the 'rooms ring camera' dataset of [28]; similar datasets were used in [35, 26], but in all cases without OOD test splits. In the training split, the camera viewpoints are on a circular path around the center of the room, with the camera pointed at the center at fixed elevation angle. 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. There are 3–4 objects present; these are placed near the side of the room identified by an odd texture. We define the following OOD test splits: (i) *position*: the 3–4 objects are now placed near a different wall; (ii) *composition*: unseen combinations of background and object textures; (iii) *number of objects*: 1/5/6 objects instead of 3–4; (iv) *viewpoint*: camera positions and elevations are randomly sampled. Exact details of the data generation process are given in the supplementary material.

ARROW. We render images using a modified version of the CLEVR dataset [46], similar to those in [45]. These have 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 (fourth) object. Object colors are randomly sampled. The camera has a random azimuth and shallow elevation. We define the following OOD test splits: (i) *position*: the four objects are positioned in a line, and the arrow no longer points to an odd object; (ii) *composition*: all objects are the same shape and color, with no arrow present; (iii) *number of objects*: 1/5/6 objects instead of four; the arrow still points at one odd object; (iv) *viewpoint*: camera looks down steeply on the objects.

4.2 Tasks and Metrics

We consider the following discriminative tasks, all taking a single image as input:

- instance segmentation, measured by adjusted Rand index (**ARI**) and mean segmentation covering (**mSC**) [26]; in the PER-SCENE setting, this measures how well the model infers segmentation maps for novel viewpoints
- depth prediction, measured by the mean relative error between predicted and true depths (**D.MRE**)
- pixel reconstruction, measured by peak signal-noise ratio (**PSNR**); in the PER-IMAGE setting, this measures faithfully our latent representation can autoencode OOD images; in the PER-SCENE setting, it measures how well the model performs novel view synthesis (NVS)

We report these metrics according to two different protocols: (i) PER-IMAGE, where we calculate the metrics only on the input image passed to the model; and (ii) PER-SCENE, where we calculate the metrics jointly over 10 images of the scene, taken from different viewpoints, but still having received only one of these as input. Thus, the latter setting measures how well the model predicts appearance, depth and segmentation from novel viewpoints. Finally, we evaluate *a priori* image generation, measured by the Fréchet Inception distance (**FID**) [36] and kernel Inception distance (**KID**) [13] between sampled and ground-truth image distributions.

4.3 Baselines

We compare our approach to three existing works. **Slot Attention [64]** is a recent unsupervised object segmentation model, with a spatial mixture representation. It is purely 2D and not generative; however, it does support generalisation to differing numbers of objects at test time. **uORF [111]** 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 [54]** is a generative method over 3D NeRFs, but which does not separate individual objects in its latent space. We used the original implementation of [64], with hyperparameters re-tuned on our datasets. We reimplemented [111, 54] in our own framework, and denote these as uORF* and NeRF-VAE* respectively. For a fair comparison, we use the same number of samples along each ray, the same or larger number of parameters, and similar encoder/decoder architectures as for our own model. Full details are in the

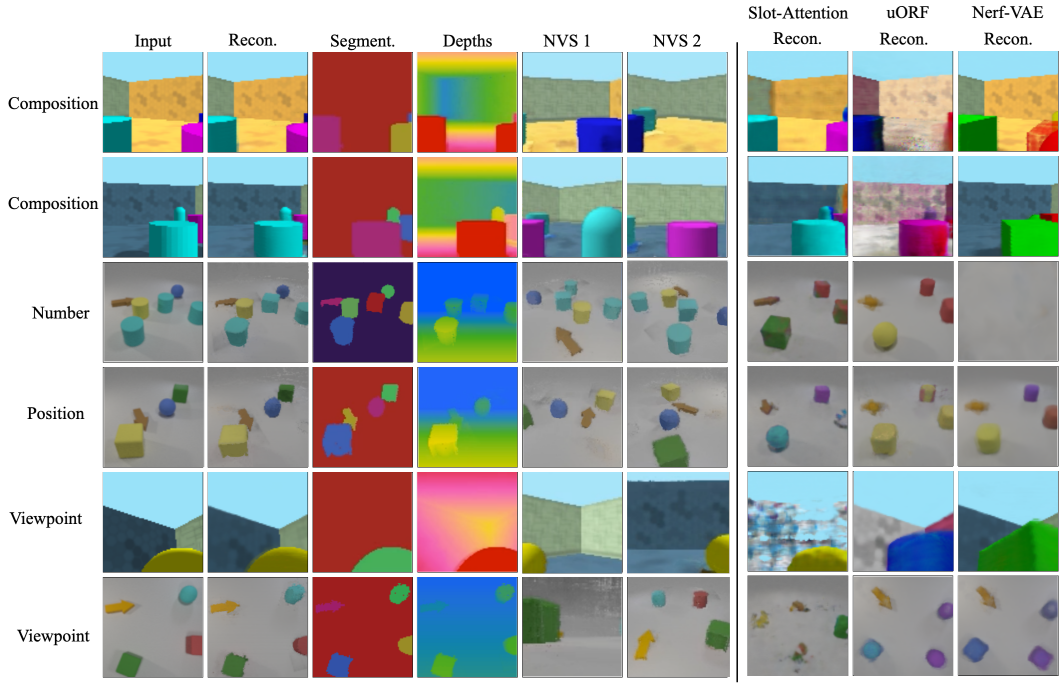

Figure 2: Qualitative results on various tasks, using our model and the baselines, on out-of-distribution (OOD) data. Each row shows the input image, then (col. 2–6) outputs from our model: the reconstruction, instance segmentation, depth map, and two novel viewpoints. We see that our model predicts high-quality segmentations and depth-maps, and that the new viewpoints are plausible and consistent with the input. The final three columns show reconstructions (the easiest task) by the baselines. Here we see that they fail to generalise to OOD data – they have specialised to their training dataset, and learnt shortcuts [29] that interpret the input images as if they were drawn from it. Thus, they fail to map objects in unfamiliar viewpoints or contexts to appropriate latent representations.

supplementary. We also modify NeRF-VAE to take just one view as input, and use non-iterative amortized inference for computational tractability.

4.4 Results

In-distribution data. We first evaluate how each model performs on the distribution of images it was trained on. Results on the discriminative 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. 4.4, and standard deviations on all results are given in the supplementary. We see that all methods successfully reconstruct input images (high per-image PSNR), with NeRF-VAE* slightly out-performing other approaches. The 3D-aware methods also predict depths with a small relative error (low per-image D.MRE), again with NeRF-VAE* best. A similar pattern holds for novel views (per-scene setting), albeit with slightly poorer scores overall. For GQN, Slot Attention performs best on segmentation under ARI scores, with ours better according to mSC and for both metrics on ARROW. Results for image generation are given in Tab. 2 and Fig. 3 (see the supplementary for images from NeRF-VAE*). Note that Slot Attention and uORF* are excluded, as they cannot generate images *a priori*. Our model has clearly learnt the distribution of both datasets, including the space of likely object and background appearances. Quantitatively, ours performs similarly to NeRF-VAE* on GQN (with FID slightly better and KID slightly worse), and out-performs it on both metrics for ARROW. In Fig. 4, we also show that our semantically-interpretable latent space allows users to edit the scene shown in an image, by counterfactual inference over the model.

OOD data. We now consider the more challenging setting where test data is drawn from a different distribution than the training set. Quantitative results in this setting 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. We see that in general, our method significantly out-performs the baselines on these splits – showing the benefit of our causal model and inference scheme. In particular, our method performs best on

	GQN		ARROW	
	FID	KID	FID	KID
Ours	80.3	0.053	141.4	0.167
NeRF-VAE*	84.2	0.047	182.7	0.190

Table 2: Quantitative results on image generation. We report FID and KID scores for our method and baseline NeRF-VAE.

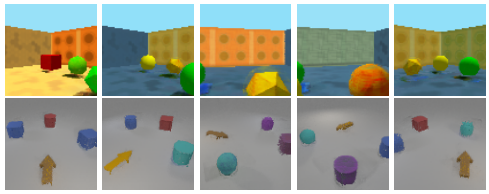

Figure 3: Examples of images sampled from our model for GQN (top) and ARROW (bottom)

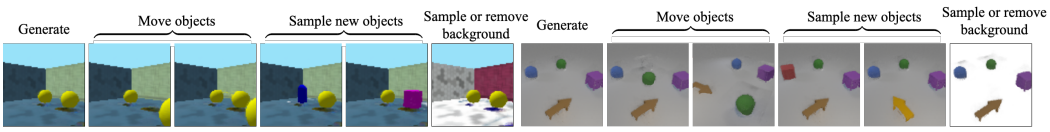

Figure 4: Editing scenes using our model, by counterfactual inference. We condition on the input scene, then modify (intervene on) certain latent variables and re-render the result

all combinations of tasks and datasets, except for depth estimation on ARROW where uORF* is slightly better. Thus, the best methods on the in-distribution test split are therefore *not* the best on OOD data. They fit the statistics of the training distribution very well, but since they do not reflect its causal structure, they learn shortcuts [29] that do not generalise well on the OOD splits. Qualitative results (Fig. 4.4) reinforce this interpretation – we see that our method still predicts accurate segmentations and depth-maps for OOD data, and novel viewpoints look plausible. Notably, in all cases, it successfully reconstructs the input image via its latent space and synthesises new viewpoints, in spite of never having seen such an image during training. In contrast, the baselines struggle with images that are different to their training distribution – 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 when they are in an unfamiliar pose (5th row, Slot Attention).

4.5 Limitations and Societal Impacts

Currently our method has the following limitations.

- The scene-level prior distribution $p_\theta(\mathbf{z}^s \mid \mathbf{z}^g)$ uses an unstructured latent space and decoder – thus, this particular component of the model is not interpretable. It would be worthwhile to use a structured prior, that explicitly models sparse relations among objects.
- We have only tested it on synthetic data; it would be interesting to conduct experiments on natural images. This would require modelling the image-formation process more faithfully, e.g. by modelling light transport and non-Lambertian materials.
- Our MCMC inference scheme is computationally expensive, as we render a complete image at each step. The efficiency could be increased by using a stochastic gradient for the LD transitions.
- Our model lacks variables to explicitly model object presence – if an object slot is unused, it learns a latent representation of a zero-density object.

We do not anticipate any negative societal impacts from this work. While all generative models have the potential to be used for creating fake content, our method requires further development to work on realistic images. On discriminative tasks, support for OOD inference should reduce the susceptibility of models to dataset bias; thus, we see a significant long-term benefit to such methods.

4.6 Conclusion

We have presented a new object-centric generative model that captures the causal process by which images are produced, and incorporates universal physical knowledge such as 3D geometry. We have shown that it is possible to intervene on our model then perform Monte-Carlo inference, in order to process out-of-distribution images. On instance segmentation and novel view synthesis in this OOD setting, it significantly out-performs three state-of-the-art approaches.

References

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

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

[3] M. A. Alcorn, Q. Li, Z. Gong, C. Wang, L. Mai, W.-S. Ku, and A. Nguyen. 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*, pages 4845–4854, 2019.

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

[5] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz. Invariant risk minimization. *CoRR*, abs/1907.02893, 2019.

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

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

[8] H. Barlow. Cerebral cortex as model builder. In *Matters of intelligence*, pages 395–406. Springer, 1987.

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

[10] Y. Bengio. The consciousness prior. *CoRR*, abs/1709.08568, 2017.

[11] J. Besag. 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.

[12] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[13] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018.

[14] R. N. Bracewell and R. N. Bracewell. *The Fourier transform and its applications*, volume 31999. McGraw-hill New York, 1986.

[15] L. Castrejon, N. Ballas, and A. Courville. INFERNO: Inferring object-centric 3d scene representations without supervision, 2022.

[16] E. R. Chan, M. Monteiro, P. Kellnhofer, J. Wu, and G. Wetzstein. 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*, pages 5799–5809, 2021.

[17] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[18] P. Dayan, G. E. Hinton, R. M. Neal, and R. S. Zemel. The helmholtz machine. *Neural computation*, 7(5):889–904, 1995.

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

[20] Y. Deng, J. Yang, J. Xiang, and X. Tong. Gram: Generative radiance manifolds for 3d-aware image generation, 2021.

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

[22] D. Driess, Z. Huang, Y. Li, R. Tedrake, and M. Toussaint. Learning multi-object dynamics with compositional neural radiance fields. *CoRR*, abs/2202.11855, 2022.

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

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

[25] M. Engelcke, O. P. Jones, and I. Posner. GENESIS-V2: inferring unordered object representations without iterative refinement. In M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 8085–8094, 2021.

[26] M. Engelcke, A. R. Kosiorek, O. P. Jones, and I. Posner. 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.

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

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

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

[30] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. 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.

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

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

[33] M. Guo, A. Fathi, J. Wu, and T. A. Funkhouser. Object-centric neural scene rendering. *CoRR*, abs/2012.08503, 2020.

[34] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, Apr. 1970.

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

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

[37] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner. 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.

[38] G. Hinton. Taking inverse graphics seriously, 2013.

[39] G. E. Hinton. How to represent part-whole hierarchies in a neural network. *CoRR*, abs/2102.12627, 2021.

[40] H. Izadinia, Q. Shan, and S. M. Seitz. Im2cad. In *CVPR*, pages 5134—-5143, 2017.

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

[42] E. Jang, S. Gu, and B. Poole. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[43] W. Jang and L. Agapito. 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*, pages 12929–12938. IEEE, 2021.

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

[45] J. Jiang and S. Ahn. Generative neurosymbolic machines. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

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

[47] R. Kabra, D. Zoran, G. Erdogan, L. Matthey, A. Creswell, M. M. Botvinick, A. Lerchner, and C. P. Burgess. Simone: View-invariant, temporally-abstracted object representations via unsupervised video decomposition. In M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 20146–20159, 2021.

[48] L. Karazija, I. Laina, and C. Rupprecht. Clevrtex: A texture-rich benchmark for unsupervised multi-object segmentation. In J. Vanschoren and S. Yeung, editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021.

[49] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[50] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In Y. Bengio and Y. LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[51] Y. Kobayashi, M. Suzuki, and Y. Matsuo. Learning global spatial information for multi-view object-centric models, 2022.

[52] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

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

[54] A. R. Kosiorek, H. Strathmann, D. Zoran, P. Moreno, R. Schneider, S. Mokrá, and D. J. Rezende. Nerf-vae: A geometry aware 3d scene generative model. In M. Meila and T. Zhang, editors, *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*, pages 5742–5752. PMLR, 2021.

[55] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25, 2012.

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

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

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

[59] J. Li, Z. Feng, Q. She, H. Ding, C. Wang, and G. H. Lee. Nemi: Unifying neural radiance fields with multiplane images for novel view synthesis. *CoRR*, abs/2103.14910, 2021.

[60] Z. Lin, Y. Wu, S. V. Peri, W. Sun, G. Singh, F. Deng, J. Jiang, and S. Ahn. 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.

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

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

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

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

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

[66] C. J. Maddison, A. Mnih, and Y. W. Teh. The concrete distribution: A continuous relaxation of discrete random variables. *CoRR*, abs/1611.00712, 2016.

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

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

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

[70] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 65(1):99–106, 2022.

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

[72] V. Nagarajan, A. Andreassen, and B. Neyshabur. 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.

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

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

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

[76] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. M. Sajjadi, A. Geiger, and N. Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. *CoRR*, abs/2112.00724, 2021.

[77] M. Niemeyer and A. Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11453–11464, 2021.

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

[79] J. Ost, F. Mannan, N. Thuerey, J. Knodt, and F. Heide. Neural scene graphs for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2856–2865, 2021.

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

[81] J. Pearl. *Causality*. Cambridge University Press, 2 edition, 2009.

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

[83] J. Peters, D. Janzing, and B. Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

[84] R. Ranganath, S. Gerrish, and D. M. Blei. Black box variational inference. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22-25, 2014*, volume 33 of *JMLR Workshop and Conference Proceedings*, pages 814–822. JMLR.org, 2014.

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

[86] M. T. Ribeiro, S. Singh, and C. Guestrin. "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*, pages 1135–1144, 2016.

[87] L. Romaszko, C. K. Williams, P. Moreno, and P. Kohli. 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*, pages 851–859, 2017.

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

[89] B. Schölkopf, D. Janzing, J. Peters, E. Sgouritsa, K. Zhang, and J. M. Mooij. 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.

[90] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio. Towards causal representation learning. *CoRR*, abs/2102.11107, 2021.

[91] L. Schott, J. von Kügelgen, F. Träuble, P. Gehler, C. Russell, M. Bethge, B. Schölkopf, F. Locatello, and W. Brendel. 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.

[92] K. Schwarz, Y. Liao, M. Niemeyer, and A. Geiger. GRAF: generative radiance fields for 3d-aware image synthesis. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

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

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

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

[96] V. Sitzmann, M. Zollhöfer, and G. Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 1119–1130, 2019.

[97] K. Stelzner, K. Kersting, and A. R. Kosiorek. Decomposing 3d scenes into objects via unsupervised volume segmentation. *CoRR*, abs/2104.01148, 2021.

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

[99] A. Trevithick and B. Yang. GRF: learning a general radiance field for 3d scene representation and rendering. *CoRR*, abs/2010.04595, 2020.

[100] S. van Steenkiste, K. Kurach, J. Schmidhuber, and S. Gelly. Investigating object compositionality in generative adversarial networks. *Neural Networks*, 130:309–325, 2020.

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

[102] J. von Kügelgen, I. Ustyuzhaninov, P. V. Gehler, M. Bethge, and B. Schölkopf. Towards causal generative scene models via competition of experts. *CoRR*, abs/2004.12906, 2020.

[103] M. Welling. Do we still need models or just more data and compute?, Apr 2019. Retrieved from `https://staff.fnwi.uva.nl/m.welling/wp-content/uploads/Model-versus-Data-AI-1.pdf` on 1 March 2022.

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

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

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

[107] Y. Xue, Y. Li, K. K. Singh, and Y. J. Lee. Giraffe hd: A high-resolution 3d-aware generative model, 2022.

[108] L. Yariv, Y. Kasten, D. Moran, M. Galun, M. Atzmon, R. Basri, and Y. Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[109] N. Ye, K. Li, L. Hong, H. Bai, Y. Chen, F. Zhou, and Z. Li. Ood-bench: Benchmarking and understanding out-of-distribution generalization datasets and algorithms. *CoRR*, abs/2106.03721, 2021.

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

[111] H.-X. Yu, L. Guibas, and J. Wu. Unsupervised discovery of object radiance fields. In *International Conference on Learning Representations*, 2022.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] see Sec. 4.5
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] see Sec. 4.5
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] The code (including for dataset generation) is proprietary; however we aim to provide enough details to reproduce
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] see Sec. 4 and supplementary
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] see supplementary; all discussion in main paper Sec. 4.4 accounts for these
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] see Sec. 4
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes] see supplementary section 'Asset Licenses'
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] code is included in supplementary, including for dataset generation
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] we use only datasets of simple synthetic environments and shapes
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]