Unsupervised Discovery of 3D Physical Objects from Video

Yilun Du MIT
Kevin Smith MIT
Tomer Ullman Harvard University
Joshua Tenenbaum MIT
Jiajun Wu Stanford University

Abstract

We study the problem of unsupervised physical object discovery. Unlike existing frameworks that aim to learn to decompose scenes into 2D segments purely based on each object's appearance, we explore how physics, especially object interactions, facilitates learning to disentangle and segment instances from raw videos, and to infer the 3D geometry and position of each object, all without supervision. Drawing inspiration from developmental psychology, our Physical Object Discovery Network (POD-Net) uses both multi-scale pixel cues and physical motion cues to accurately segment observable and partially occluded objects of varying sizes, and infer properties of those objects. Our model reliably segments objects on both synthetic and real scenes. The discovered object properties can also be used to reason about physical events.

1 Introduction

From early in development, infants impose structure on their world. When they look at a scene, infants do not perceive simply an array of colors. Instead, they scan the scene and organize the world into objects that obey certain physical expectations, like traveling along smooth paths or not winking in and out of existence [22, 23]. Here we take two ideas from human, and particularly infant, perception for helping artificial agents learn about object properties: that coherent object motion constrains expectations about future object states, and that foveation patterns allow people to scan both small or far-away and large or close-up objects in the same scene.

Motion is particularly crucial in the early ability to segment a scene into individual objects. For instance, infants perceive two patches moving together as a single object, even though they look perceptually distinct to adults [11]. This segmentation from motion even leads young children to expect that if a toy resting on a block is picked up, both the block and the toy will move up as if they are a single object. This suggests that artificial systems that learn to segment the world could be usefully constrained by the principle that there are objects that move in regular ways.

In addition, human vision exhibits foveation patterns, where only a local patch of a scene is often visible at once. This allows people to focus on objects that are otherwise small on the retina, but also stitch together different glimpses of larger objects into a coherent whole.

We propose the Physical Object Discovery Network (POD-Net), a self-supervised model that learns to extract object-based scene representations from videos using motion cues. POD-Net links a visual generative model with a dynamics model in which objects persist and move smoothly. The visual generative model factors an object-based scene decompositions across local patches, then aggregates those local patches into a global segmentation. The link between the visual model and the dynamics model constrains the discovered representations to be usable to predict future world states. POD-Net thus produces more stable image segmentations than other self-supervised segmentation models, especially in challenging conditions such as when objects are close together or occlude each other (Figure 1).

Preprint. Under review.

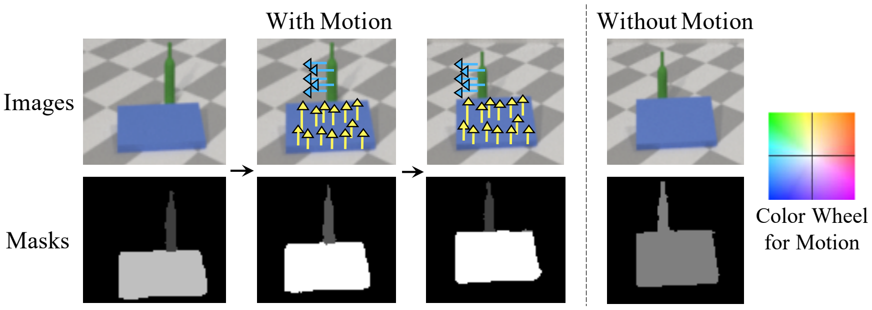


Figure 1: Motion is an important cue for object segmentation from early in development. We combine motion with an approximate understanding of physics to discover 3D objects that are physically consistent across time. In the video above, motion cues (shown with colored arrows) enable our model to modify our predictions from a single large incorrect segmentation mask to two smaller correct masks.

We test how well POD-Net performs image segmentation and object discovery on two datasets: one made from ShapeNet objects [2], and one from real-world images. We find that POD-Net outperforms recent self-supervised image segmentation models that use regular foreground-background relationships [7] or assume that images are composable into object-like parts [1]. Finally, we show that the representations learned by POD-Net can be used to support reasoning in a task that requires identifying scenes with physically implausible events [21]. Together, this demonstrates that using motion as a grouping cue to constrain the learning of object segmentations and representations achieves both goals: it produces better image segmentations and learns scene representations that are useful for physical reasoning.

Related work. Developing a factorized scene representation has been a core research topic in computer vision for decades. Most learning-based prior works are supervised, requiring annotated specification such as segmentations [9], patches [5], or simulation engines [27, 10]. These supervised approaches face two challenges. First, in practical scenarios, annotations are often prohibitively challenging to obtain: we cannot annotate the 3D geometry, pose, and semantics of every object we encounter, especially for deformable objects such as trees. Second, supervised methods may not generalize well to out-of-distribution test data such as novel objects or scenes.

Recent research on unsupervised object discovery and segmentation has attempted to address these issues: researchers have developed deep nets and inference algorithms that learn to ground visual entities with factorized generative models of static [6, 1, 7, 3] and dynamic [25, 26, 14, 4] scenes. Some approaches also learn to model the relations and interactions between objects [26, 24, 25]. The progress in the field is impressive, though these approaches are still mostly restricted to low-resolution images and perform less well on small or heavily occluded objects. Because of this, they often fail to observe key concepts such as object permanence and solidity. Further, these models all segment objects in 2D, while our POD-Net aims to capture the 3D geometry of objects in the scene.

Some recent papers have integrated deep learning with differentiable rendering to reconstruct 3D shapes from visual data without supervision, though they mostly focused on images of a single object [18, 20], or require multi-view data as input [28]. In contrast, we use object motion and physics to discover objects in 3D with physical occupancy. This allows our model to do better in both object discovery and future prediction, captures notions such as object permanence, and better aligns with people's perception, belief, and surprise signals of dynamic scenes.

2 Method

The Physical Object Discovery Network (POD-Net) (Figure 2) decomposes a dynamic scene into a set of component 3D physical primitives. POD-Net contains an inference model, which recursively infers a set of component primitive descriptions, masks, and latent vectors (Section 2.1). It also contains a three-module generative model (Section 2.2). The generative model uses a back-projection module to infer 3D properties of each component. It also includes a dynamics model to predict primitives motions and a VAE [12, 17] to back-project these primitives onto 2D images. These components ensure that the learned primitive representations can reconstruct the original image, as

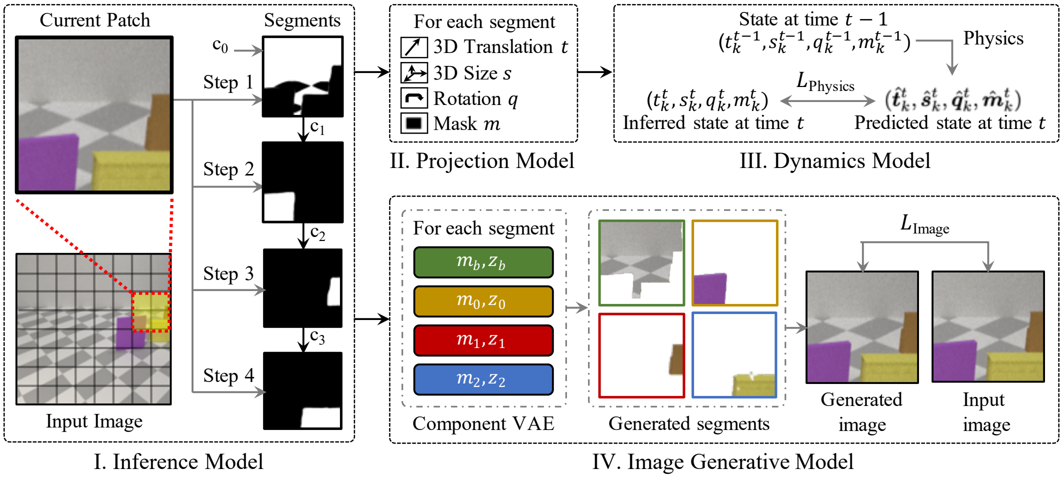


Figure 2: POD-Net contains four modules for discovering physical objects from video. (I) An inference model auto-regressively infers a set of candidate object masks and latents to describe each patch of an image; (II) A projection model maps each mask to a 3D primitive; (III) A dynamics model captures the motion of 3D physical objects; and (IV) An image generative model decodes proposed latents and masks to reconstruct the image.

well as a sequence of images consistent with physical dynamics. Together, these constraints produce a strong signal for self-supervised learning of object-centric scene representations.

2.1 Inference Model

We sequentially infer the underlying masks and latents that represent a scene (Figure 2-I). Inspired by MONet [1], we employ an attention network to iteratively decompose a scene into a set of separate masks $\boldsymbol{M} = \{\boldsymbol{m}_1, \boldsymbol{m}_2, ..., \boldsymbol{m}_n\}$. For each mask $\boldsymbol{m}_i$, a corresponding latent vector $\boldsymbol{z}_i$ is extracted.

In particular, we initialize context $\boldsymbol{c}_0 = \mathbf{1}$, which we define to represent the context in the image $\boldsymbol{x}$ yet to be explained. At each step, we decode the attention mask $\boldsymbol{m}_i = \boldsymbol{c}_{i-1}\alpha_\psi(\boldsymbol{x}; \boldsymbol{c}_{i-1})$, using a parameterized attention network Attention$(\cdot)$. We iteratively update the corresponding context in the image by $\boldsymbol{c}_i = \boldsymbol{c}_{i-1}\left(1 - \alpha_\psi(\boldsymbol{x}; \boldsymbol{c}_{i-1})\right)$ to ensure that sum of all masks explain the entire image.

We further train a VAE encoder Encode$(z|m, \boldsymbol{x})$, which infers latents $\boldsymbol{z}_i$ from each component mask $\boldsymbol{m}_i$. We set $\boldsymbol{m}_0, \boldsymbol{z}_0$ – the first decoded mask and latent – to be the background mask $\boldsymbol{m}_b$ and latent $\boldsymbol{z}_b$, and define each subsequent mask or latent to be object masks and latents.

Sub-patch decomposition. Direct inference of component objects and background from a single image can be difficult, especially when images are complex and when objects are of vastly different sizes. An inference network must learn to pay attention to coarse features in order to segment large objects, and to fine details in the same image order to segment the smaller objects. Inspired by how people solve this problem by stitching together multiple foveations into a coherent whole, we train our models and apply inference on overlapping sub-patches of an image (Figure 3).

In particular, given an image of size $H \times W$, we divide into the image into a 8×8 grid (pictured in the left of Figure 3), with each grid element having size $H/8 \times W/8$. We construct a sub-patch for every 2×2 component subgrid, leading to a total of 64 different overlapping subpatches. We apply inference on each sub-patch. Under this decomposition, smaller objects still appear large in each sub-patch, while larger objects are shared across sub-patch.

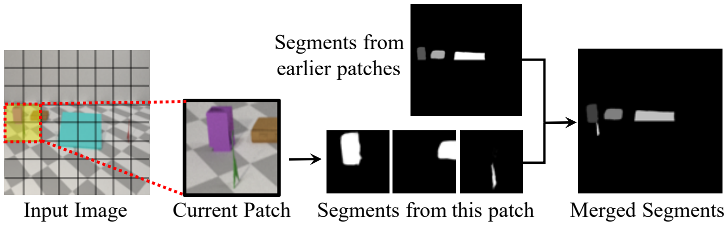


Figure 3: Illustration of sub-patch decomposition for image inference. An image is divided in a 8 by 8 grid, with inference is applied to each 2 by 2 subgrid. To generate a global segmentation mask, object masks are sequentially inferred for each subpatch. Each object mask is either matched to an existing object or used to create a new object.

To obtain a global segmentation map, we merge each sub-patch sequentially using a sliding window (Figure 3). At each step, we iterate through each segment given by the inference model from a sub-patch, and merge it with segments obtained from previous sub-patches, if there is an overlap in masks above 20 pixels. Every segment that does not get merged is initialized as a new object.

2.2 Generative Model

Our generative model represents a dynamic scene as a set of K different physical objects and the surrounding background at each time step t. Each physical object k is represented by its back-projection on 2D, a segmentation mask $\boldsymbol{m}_k^t \in \mathbb{R}^{HxW}$ of height H and width W, and a latent code $\boldsymbol{z}_k^t \in \mathbb{R}^D$ of dimension D for its appearance. In addition, the background is captured as a surrounding segmentation mask $\boldsymbol{m}_b^t \in \mathbb{R}^{HxW}$ and code $\boldsymbol{z}_b^t \in \mathbb{R}^D$. Segmentation masks are defined such that the sum of all masks correspond to the entire image $\sum_k \boldsymbol{m}_k^t + \boldsymbol{m}_b^t = 1$.

We use a projection model to map segmentation masks $\boldsymbol{m}_k^t$ to 3D primitive cuboids (Figure 2-II). Cuboids are a coarse geometric representation that enable physical simulation. We next construct a dynamics model over the physical movement of predicted primitives (Figure 2-III). We further construct a generative model over images $\boldsymbol{x}^t$ by decoding latents $\boldsymbol{z}^t$ component-wise (Figure 2-IV).

Projection model. Our projection model maps a mask $\boldsymbol{m}_k$ to an underlying 3D primitive cuboid, represented as a translation $\boldsymbol{t}_k \in \mathbb{R}^3$, size $\boldsymbol{s}_k \in \mathbb{R}^3$, and rotation $\boldsymbol{q}_k \in \mathbb{R}^3$ (as a Euler angle) transform on a unit cuboid in a fully differentiable manner. This task can be done by assuming the camera parameters and the height of the plane is given. In our case, we pre-train a neural network to approximate the 2D-to-3D projection and use it as our differentiable projection model.

Dynamics model. We construct a dynamics model over the next state of different physical objects $(\boldsymbol{t}_k^t, \boldsymbol{s}_k^t, \boldsymbol{q}_k^t, \boldsymbol{m}_k^t)$ by using first order approximation of velocity/angular velocity of the states of the object. Specifically, our model predicts

$$\hat{\boldsymbol{t}}_k^t = \boldsymbol{t}_k^{t-1} + \frac{1}{t-1}\sum_{i=1}^{t-1}(\boldsymbol{t}_k^i - \boldsymbol{t}_k^{i-1}), \quad \hat{\boldsymbol{s}}_k^t = \frac{1}{t}\sum_{i=0}^{t-1}\boldsymbol{s}_k^i \tag{1}$$

$$\hat{\boldsymbol{q}}_k^t = \boldsymbol{q}_k^{t-1} + \frac{1}{t-1}\sum_{i=1}^{t-1}(\boldsymbol{q}_k^i - \boldsymbol{q}_k^{i-1}), \quad \hat{\boldsymbol{m}}_k^t = \text{Render}(\hat{\boldsymbol{t}}_k^t, \hat{\boldsymbol{s}}_k^t, \hat{\boldsymbol{q}}_k^t). \tag{2}$$

The Render function is defined as $\text{Render}(\hat{\boldsymbol{t}}_k^t, \hat{\boldsymbol{s}}_k^t, \hat{\boldsymbol{q}}_k^t) = \mathbb{1}_{\text{foreground}}\text{UnProject}(\hat{\boldsymbol{t}}_k^t, \hat{\boldsymbol{s}}_k^t, \hat{\boldsymbol{q}}_k^t)$, where $\text{UnProject}(\cdot)$ is a pre-trained model that projects each primitive in 3D to a 2D segmentation mask (inverse of the Projection model described above). $\mathbb{1}_{\text{foreground}}$ is an indicator function of whether the object is at the foreground and visible, and equals to 1 when a $\hat{\boldsymbol{t}}_k^t$ is closer than all other objects at the specified pixel location.

Given modeled future states, the overall likelihood of a physical object $(\boldsymbol{t}_k^t, \boldsymbol{s}_k^t, \boldsymbol{q}_k^t, \boldsymbol{m}_k^t)$ is given by

$$p(\boldsymbol{t}_k^t, \boldsymbol{s}_k^t, \boldsymbol{q}_k^t, \boldsymbol{m}_k^t) = \mathcal{N}(\boldsymbol{t}_k^t; \hat{\boldsymbol{t}}_k^t, \sigma^2)\mathcal{N}(\boldsymbol{s}_k^t; \hat{\boldsymbol{s}}_k^t, \sigma^2)\mathcal{N}(\boldsymbol{q}_k^t; \hat{\boldsymbol{q}}_k^t, \sigma^2)p(\hat{\boldsymbol{m}}_k^t, \boldsymbol{m}_k^t), \tag{3}$$

where we assume a Gaussian distributions over translation, sizes, and rotations with $\sigma = 1$. $p(\cdot)$ is the probability of a predicted mask, defined as

$$p(\hat{\boldsymbol{m}}_k^t, \boldsymbol{m}_k^t) = \left[\mathbb{1}_{\boldsymbol{m}_k^t > 0.5}\hat{\boldsymbol{m}}_k^t\right]\left[\mathbb{1}_{\hat{\boldsymbol{m}}_k^t > 0.5}\boldsymbol{m}_k^t\right], \tag{4}$$

where $\mathbb{1}_(\cdot)$ is the indicator function. Note that $p(\cdot)$ is a differentiable expression that encourages both $(\hat{\boldsymbol{m}}_k^t, \boldsymbol{m}_k^t)$ to be similar to each other.

Image generative model. We represent images $\boldsymbol{x}^t$ at each time step as spatial Gaussian mixture models. Each latent $\boldsymbol{z}_k$ is decoded to a pixel-wise mean μ_k and a predicted mask $\boldsymbol{c}_k$ using a VAE decoder $\text{Decode}(\mu_k, \boldsymbol{c}_k | \boldsymbol{z}_k)$ [13]. We assume each pixel i is independent conditioned on $\boldsymbol{z}$, so that the likelihood becomes

$$p(\boldsymbol{x}|\boldsymbol{z}) = \sum_{k=1}^{K}(m_i\mathcal{N}(x_i; \mu_i, \sigma^2) + p(\boldsymbol{c}_i|m_i)) + m_b\mathcal{N}(x_i; \mu_b, \sigma_b^2) + p(\boldsymbol{c}_b|m_b) \tag{5}$$

for background component m_b, μ_b, c_b and object components m_i, μ_i, c_i. We use $\sigma = 0.11$ and $\sigma_b = 0.07$ to break symmetry between object and background components, encouraging the background to model the more uniform image components [1]. Our overall loss encourages the decomposition of an image into a set of reusable sub-components, as well as a large, uniform background.

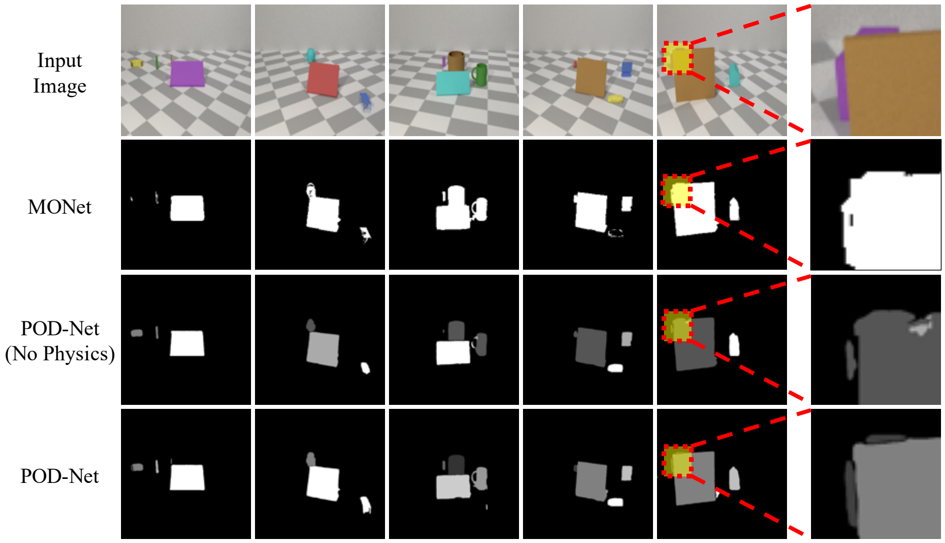


Figure 4: Comparisons of unsupervised object segmentation of POD-Net with and without motion and with MONet on scenes with synthetic objects. MONet is unable to seperate individual instances of objects, but is capable of getting a foreground mask of objects in a scene. POD-Net (no physics) is able to reliably detect almost all objects, though some instances of objects are merged together into a single object. POD-Net is able to reliably detect separate objects even when they are mostly occluded (zoomed-in images on right).

2.3 Training Loss

Our overall system is trained to maximize the likelihood of both physical object and image generative models. Our loss consists of $\mathcal{L}(\theta_{\text{attn}}, \theta_{\text{enc}}, \theta_{\text{dec}}, \boldsymbol{x}^t) = \mathcal{L}_{\text{Physics}} + \mathcal{L}_{\text{Image}} + \mathcal{L}_{\text{KL}}$, maximizing the likelihood of physical dynamics, images, and variational bound. Our image loss is defined to be

$$\mathcal{L}_{\text{Image}} = -\sum_{k=1}^{K} (\boldsymbol{m}_k^t \log(\text{Decode}(\boldsymbol{x}^t|\boldsymbol{z}_k^t)) + \log(\text{Decode}(\boldsymbol{m}_k^t|\boldsymbol{z}_k^t))), \tag{6}$$

enforcing that latents decode to corresponding object and background component masks and values. Our physics loss is defined to be

$$\mathcal{L}_{\text{Physics}} = -\sum_{k=1}^{K} \log p(\boldsymbol{t}_k^t, \boldsymbol{s}_k^t, \boldsymbol{q}_k^t, \boldsymbol{m}_k^t), \tag{7}$$

which enforces that decoded primitives are physically consistent. And the KL loss is defined as

$$\mathcal{L}_{\text{KL}} = \beta(\sum_{k=1}^{K} D_{KL}(\text{Encode}(\boldsymbol{z}_k^t|\boldsymbol{x}^t, \boldsymbol{m}_k^t) \mid\mid p(z)) + D_{KL}(\text{Encode}(\boldsymbol{z}_b^t|\boldsymbol{x}^t, \boldsymbol{m}_b^t) \mid\mid p(z))) \tag{8}$$

to enforce the variational lower bound on likelihood [13] for latents inferred on both background and foreground components.

Our training paradigm consists of two different steps. We first maximize the likelihood of the model under the image generation objective. After qualitatively observing object like masks, we switch to maximize the likelihood of the model under both the generation and physical plausibility objective. We find that enforcing physical consistency during early stages of training detrimental, as the model has not discovered object like primitives yet. We use the RMSprop optimizer with a learning rate of 10^{-4} within the PyTorch framework [16] to train our models.

3 Evaluation

We evaluate POD-Net on unsupervised object discovery in two different scenarios: a synthetic data set consisting of various moving ShapeNet objects, and a real dataset of block towers falling. We also test how inferred 3D primitives can support more advanced physical reasoning.

3.1 Moving ShapeNet

We use ShapeNet objects to explore the ability of POD-Net to learn to segment objects from appearance and motion cues. We also test its ability to generalize to new shapes and textures.

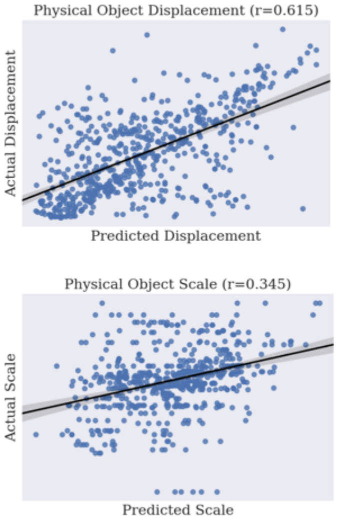


Figure 5: Plot of predicted translation of 3D primitive vs ground truth translation of 3D primitives (top) and plot of predicted scale of 3D primitive vs ground scale of 3D primitive.

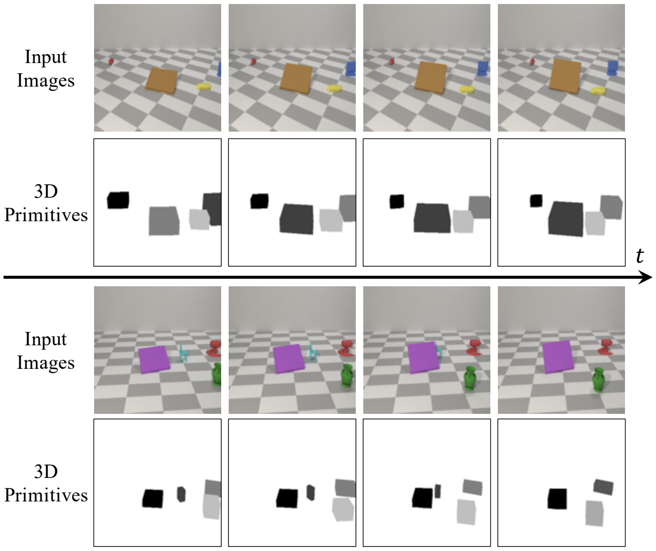


Figure 6: Visualization of discovered 3D primitive in two different scenes (top and bottom) through time. Our model is able to discover a 3D shape, that is consistent with observed inputs under a perspective map. Furthermore, discovered primitive move coherently through time.

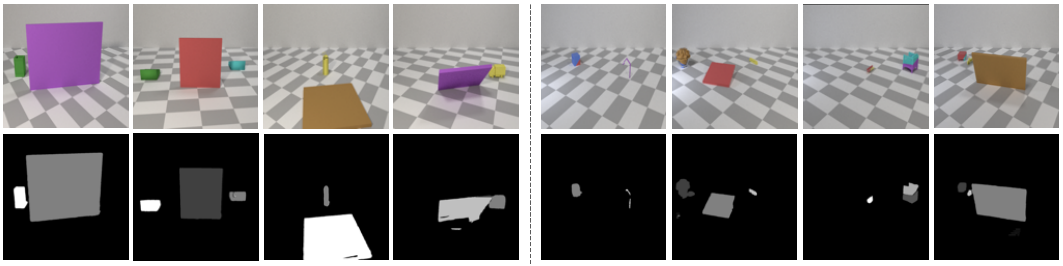
Data. To train models on moving ShapeNet objects, we use the generation code provided in the ADEPT dataset in Smith et al. [21]. We generate a training set of 1,000 videos, each 100 frames long, of objects (80% of the objects from 44 ShapeNet categories) as well as rectangular occluders. Objects move in either a straight line, back and forth, or rotate, but do not collide with each other.

Setup. The videos have a resolution of 1024×1024 pixels. We apply our model with a patch size of 256×256. We use a residual architecture [8] for the attention and VAE components. We pre-train our projection model on scenes of a single ShapeNet object, varied across different locations on a plane, with different rotations, translations, and scales. The projection model learns to map from a segmentation mask of an object to corresponding rotation, translation, and scale parameters. We note that these ShapeNet scenes are rendered using different camera extrinsics/intrinsics then those used to generate the ADEPT dataset. Furthermore, segmentation masks are never partially occluded, unlike the ADEPT dataset. Thus, the projection model just serves a rough relative map from 2D mask to corresponding 3D position/size. More details can be found in the Supplementary Material.

To the compute the physical plausibility $L_{physics}$ (Equation 7) of primitives, we utilize the observations from the last three time steps. For efficiency, we evaluate physical plausibility on each component sub-patch of image. We train a recurrent model with a total of 5 slots for each image.

Metrics. To quantify our results, we use intersection of union (IoU) between predicted segmentation masks for each object and the corresponding ground truth masks. We compute the IoU for each ground truth mask, by finding the IoU of the predicted segmentation mask with IoU with the ground truth mask. We report the average IoU across all objects in an image, as well as the percentage of objects detected in an image (with IoU $>$ 0.5).

Baselines. We compare with two recent models of self-supervised object discovery: OP3 [26] and MONet [1]. The OP3 model uses an iterative inference procedure to obtain object masks and representations through time. We use 7 slots to train the OP3 model with 4 steps of optimization per mask on the first image, and an additional step of optimization per future time step. Due to memory constraints, we were only able to train the OP3 model on inputs of size 128 by 128, using the provided codebase. The MONet model uses recurrent inference procedure to obtain object mask and representations per time step, similar to our model. In contrast to the MONet, we use a residual backbone with a different encoding of spatial coordinates, which we detail in the Supplemental Material. We train MONet on inputs of size 256 by 256. We also compare with ablations of POD-Net: applying POD-Net directly on an image (single-scale) as opposed to across patches (multi-scale), and POD-Net without physical consistency.



Generalization to New Objects

Generalization to New Colors

Model	Phys	IoU	Detection (%)
POD-Net	No	0.768	0.823
	Yes	0.857	0.922

Model	Phys	IoU	Detection (%)
POD-Net	No	0.658	0.716
	Yes	0.756	0.843

Figure 7: Generalization to novel objects and colors. Top: POD-Net successfully segments individual objects, except when colors bisect an object (row 2, column 7). Bottom: Evaluation of POD-Net's generalization with or without physical constancy, measured in average IoUs on segmentations and in the percentage of objects that are detected. Including physics integrates the motion signal and generalizes better in both cases.

Results. We quantitatively compare object masks discovered by our model and other baselines in Table 1. We find that OP3 performs poorly, as it only discovers a limited subset of objects. MONet performs better and is able to discover a single foreground mask of all objects. However, the masks are not decomposed into separate component objects in a scene (Figure 4, 2nd row). Our scenes consist of a variable set of objects of vastly different scales, making it hard for MONet to learn to assign individual slots for each object.

Table 1: Average IoU on segmentations on the ADEPT dataset and the percentage of objects detected, where at least one segmentation mask has greater than 0.5 IoU. Standard error in parentheses.

Model	**Multi-Scale**	**Phys**	**IoU**	**Detection (%)**
MONET	-	-	0.289 (0.007)	0.306 (0.005)
OP3	-	-	0.145 (0.004)	0.121 (0.007)
POD-Net	No	No	0.314 (0.010)	0.361 (0.007)
POD-Net	No	Yes	0.462 (0.007)	0.512 (0.009)
POD-Net	Yes	No	0.649 (0.011)	0.709 (0.016)
POD-Net	Yes	Yes	**0.739 (0.011)**	**0.821 (0.015)**

We find that applying POD-Net (single scale, no physics) improves on MONet slightly, discovering several different masks containing multiple objects, albeit sometime missing objects such as the occluder. POD-Net (single scale, physics) is able to more reliably able to segment separate objects, but still encounters issues of missing objects. POD-Net (multi scale, no physics) is able to reliably segment all objects in scene, but often merges multiple objects into one object, especially when objects are overlapping (e.g., Figure 4, 3rd row). Finally, POD-Net obtains the best performance and is able to segment all objects in the scene and individual objects where multiple objects overlap with each other (Fig. 4, 4th row). The full model still sometimes exhibits over-segmentation or segments sharp shadows as a separate object.

We analyze the 3D objects discovered by POD-Net. Figure 5 shows a plot of predicted displacements of discovered 3D objects with ground-truth object displacements. It also shows a plot of the predicted scale of discovered 3D objects with ground truth. The 3D objects found by POD-Net have good correlation with ground truth 3D object annotations. Visualizations of discovered objects in Figure 6 show that POD-Net is able to segment a scene into a set of 3D cuboid primitives that correspond to the objects in a video, and that these objects move consistently through time.

Generalization. Just as young children can detect and reason about new objects with arbitrary shapes and colors, we test how well POD-Net can generalize to scenes with both novel objects and colors. We evaluate the generalization of our model on two datasets.

- Novel objects: We use the test set in Smith et al. [21], consisting of the 20% novel objects from 44 ShapeNet categories, objects from another 11 ShapeNet categories not in the training dataset, and common developmental psychology objects such as toy ducks.
- Novel colors: We generated a dataset with object distribution the same as the original video dataset, but each object is split into two separate colors.

Figure 7 shows quantitative analysis of POD-Net applied to datasets with both novel objects and colors. We find that in both settings, POD-Net with physical consistency gets better segmentation than without. Numbers here are higher than those on the training set, because both novel datasets contain fewer objects in a single scene. Qualitatively, POD-Net performs well when asked to discover novel objects, though it can mistake a multi-colored novel shape to be two objects.

3.2 Real Block Towers

Next we evaluate how POD-Net segments and detects objects in real videos.

Data. We use the dataset in Lerer et al. [15] with 492 videos of real block towers, which may or may not be falling. Each frame contains 2 to 4 blocks of red, yellow, or green color. Each block has the same 3D shape, though the 2D projections on the camera differ.

Setup. For our projection model, we use a pretrained neural network on scenes of a single block at different heights, sizes, and varying distances (to account for differences in relative distance of a falling block). Similar to Section 3.1, the projection model is trained with different camera and perspective parameters than those in the data set. Furthermore, the trained dataset does not contain occlusion like the block dataset does. All other settings are the same as that used in Section 3.1.

Results. We compare masks discovered by POD-Net and baselines in Figure 8. We found that OP3 often groups multiple blocks together and misses some blocks. MONet performs better, but often misses blocks and also groups two blocks as a single object, leading to floating blocks in the air (Figure 8, 2nd row). POD-Net (single scale, no physics) is able to segment all blocks, but treats the entire stack as a single object. POD-Net (multi scale, no physics) does better and is able to reliably segment all blocks, though it still groups blocks of similar colors together (Figure 8, 3rd row). Finally, POD-Net with multiple scales and physical consistency performs the best, reliably separating individual blocks in a tower (Figure 8, 4th row).

Model	Multi-Scale	Phys	IoU	Detection (%)
MONET	-	-	0.521 (0.005)	0.537 (0.003)
OP3	-	-	0.311 (0.004)	0.250 (0.007)
POD-Net	No	No	0.546 (0.004)	0.523 (0.006)
POD-Net	Yes	No	0.734 (0.012)	0.761 (0.008)
POD-Net	Yes	Yes	**0.837 (0.004)**	**0.908 (0.008)**

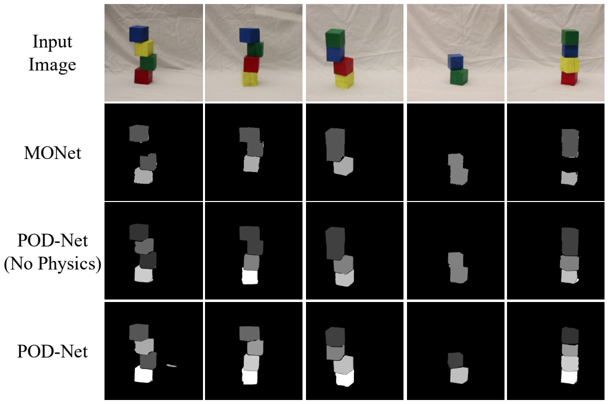


Figure 8: Top: IoU of segmentation results on the real blocks dataset and the percentage of objects detected. Bottom: Qualitative comparisons of unsupervised object segmentation of POD-Net with and without physics and with MONet on realistic block towers. MONet often groups two blocks of similar color (dark blue/green) together and sometimes misses particular blocks. POD-Net without physics reliably detects all blocks, but still groups similar blocks (dark blue/green) into one. POD-Net with physics detects all objects and assigns different masks to each. Standard error in parentheses.

3.3 Judging Physical Plausibility

We test whether POD-Net can discover objects reliably enough to perform the physical violation detection task of Smith et al. [21], in which videos that have non-physical events (objects disappearing or teleporting) must be differentiated from plausible videos.

Data. Smith et al. [21] introduced a test set of videos representing common psychologically surprising scenes to humans. Such scenes evaluate core object properties such as permanence (objects do not appear or disappear for no reason), continuity (objects move along connected trajectories), and solidity (objects can not move through each other). To test a combination of all these concepts, we evaluate how well a model with POD-Net in the loop performs prediction on the ‘Overturn (Long)’ and ‘Block’ tasks in the ADEPT benchmark.

The Overturn (Long) task consists a plane overlaying an object, requiring reasoning of object permanence. The Block task consists, one of the hardest tasks in ADEPT benchmarks, consists of physical scenes with a solid wall and a object moving towards the wall. Once the object is occluded, it may either appear to hit the wall and stop, or appear on the other side. To accomplish these tasks, a system must remember object states across a large number of time steps and understand both spatial continuity and object permanence.

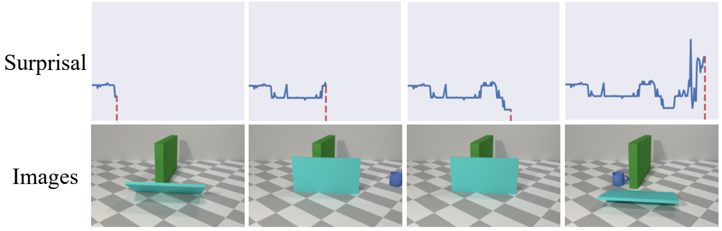


Figure 9: Model surprisal over time in a 'Block' scene. POD-Net has relatively low surprisal throughout most of the video. But when the occluder falls and the object appears to 'teleport' across the wall, POD-Net recognizes this abnormal shift in position and becomes surprised.

Setup. We use POD-Net trained in Section 3.1 to obtain a set of physical objects (represented as cuboids) describing an underlying scene. Since our approach is unsupervised, we further fine-tune POD-Net on plausible videos in block task for one thousand training iterations. The extracted objects are provided as a scene description to the stochastic physics engine described in Smith et al. [21]. We use a particle filter to maintain a set of beliefs states over the physical objects, and measure surprisal between current observations from POD-Net with those in the belief state following Smith et al. [21].

To evaluate the performance of our model, we use a relative accuracy metric [19]: given n pairs of videos with surprising scenes $\mathbf{x}^+$ and control scenes $\mathbf{x}^-$, we report the proportion of correctly ordered scene pairs such that the violation scene is judged more surprising than a matched control scene without a violation $\sum_{i,j}[c(\mathbf{x}_i^+) > c(\mathbf{x}_j^-)]/n$. We evaluate our model on 189 scene pairs.

Results. On the Block task, we find that our model achieves a relative accuracy of 0.622. Its performance on a single video can be seen in Figure 9, where it has learned to localize the block well enough that the model is surprised when it appears on the other side of the wall. The model in Smith et al. [21] scores a relative accuracy of 0.680. It acts as an upper bound for the performance of our model, since they use supervised training for discovering the object masks and recovering object properties. In contrast, POD-Net discovers 3D objects in an unsupervised manner, outperforming the baseline generative models studied by Smith et al. [21] that do not encode biases for objecthood (GAN: 0.44, Encoder-Decoder: 0.52, LSTM: 0.44).

On the Overturn (Long) task, our model obtains a performance of 0.77 compared to the 0.73 in Smith et al. [21], and models that do not encode biases for objects (GAN: 0.81, Encoder-Decoder: 0.61, LSTM: 0.63).

A current limitation of our approach towards discovering 3D object primitives is that across a long video (over 100 timesteps), there may be several spurious extraneous objects discovered. The model in Smith et al. [21] does not deal well with such spurious detections, requiring us to tune separate hyper-parameters for each task. Future work can circumvent this issue by adding a perceptual uncertainty model into Smith et al. [21].

4 Conclusion

We have proposed POD-Net, a model that discovers 3D physical objects from video using self-supervision. We show that by retaining principles of core knowledge in our architecture – that objects exist and move smoothly – and by factorizing object segmentation across sub-patches, we can learn to segment and discover objects in a generalizable fashion. We further show how these discovered objects can be utilized in downstream tasks to judge physical plausibility. We believe further exploration in this direction is a promising approach towards more robust object discovery and a richer physical understanding of the world around us.

Broader Impact

Our work is a step towards the broad goal of building a system with physical understanding of the scenes around it. A system with a broad physical understanding of its surrounding environment has many potential impacts in industrial domains such as enabling household robots that can help the elderly age in place by helping with household chores and monitoring medical treatment. On the other hand, errors in our scene understanding system, such as classifying an unstable multi-object structure as a single object, could be costly if the model were actually deployed. We do not foresee our model, as a preliminary research effort, to have any significant societal impact toward any particular group.

A.1 Appendix

A.1.1 Model Architecture

We detail our attention model in Table A1a and our component VAE model in Table A1b. In contrast to Burgess et al. [1], we use a residual architecture for both attention and component VAE networks, with up-sampling of the spatial broadcast layer.

7x7 Conv2D, 32
BatchNorm
3x3 Max Pool (Stride 2)
ResBlock Down 32
ResBlock Down 64
ResBlock Down 128
ResBlock Up 256
ResBlock Up 128
ResBlock Up 64
ResBlock Up 32
ResBlock Up 32
3x3 Conv2D, Output Channels

(a) Attention Model (α_ψ)

7x7 Conv2D, 32
BatchNorm
3x3 Max Pool (Stride 2)
ResBlock Down 16
ResBlock Down 32
ResBlock Down 64
Global Average Pool
Dense $\rightarrow$ 256
256 $\rightarrow$ 32 (μ, σ)
z $\leftarrow \mathcal{N}(\mu, \sigma)$
Spatial Broadcast z (8x)
3x3 Conv2d, 256
ResBlock up 128
ResBlock up 64
ResBlock up 32
ResBlock up 16
ResBlock up 16
3x3 Conv2D, Output Channels

(b) VAE Component Model. (q_ϕ, p_θ)

Figure A1: Overall Model Architectures used in POD-Net

A.1.2 Source Code

We attach anonymous source code used to train models in the CMT submission portal.

A.1.3 Comparison on Partially Occluded Objects

We further explicitly compare the performance of POD-Net on segmenting objects that occlude each other. We evaluate on the ADEPT dataset, but only consider objects such that the bounding boxes intersect. We find that in this dataset of objects, POD-Net (multi-scale, physics) obtains has a detection rate of 0.734, with the an average IoU of 0.701 while POD-Net (multi-scale, no physics) obtains a detection rate of 0.601 (IoU threshold 0.5) with an average IoU of 0.576. This indicates our approach in incorporating physics is able to learn to effectively separate objects that partially occlude each other.

References

[1] Christopher P Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *arXiv:1901.11390*, 2019.

[2] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *arXiv:1512.03012*, 2015.

[3] SM Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, Koray Kavukcuoglu, and Geoffrey E Hinton. Attend, infer, repeat: Fast scene understanding with generative models. In *NeurIPS*, 2016.

[4] SM Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S Morcos, Marta Garnelo, Avraham Ruderman, Andrei A Rusu, Ivo Danihelka, Karol Gregor, et al. Neural scene representation and rendering. *Science*, 360(6394):1204–1210, 2018.

[5] Katerina Fragkiadaki, Pablo Arbelaez, Panna Felsen, and Jitendra Malik. Learning to segment moving objects in videos. In *CVPR*, 2015.

[6] Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Neural expectation maximization. In *NeurIPS*, 2017.

[7] Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Chris Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-Object Representation Learning with Iterative Variational Inference. In *ICML*, 2019.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2015.

[9] Michael Janner, Sergey Levine, William T Freeman, Joshua B Tenenbaum, Chelsea Finn, and Jiajun Wu. Reasoning about physical interactions with object-oriented prediction and planning. In *ICLR*, 2018.

[10] Ken Kansky, Tom Silver, David A Mély, Mohamed Eldawy, Miguel Lázaro-Gredilla, Xinghua Lou, Nimrod Dorfman, Szymon Sidor, Scott Phoenix, and Dileep George. Schema networks: Zero-shot transfer with a generative causal model of intuitive physics. In *ICML*, 2017.

[11] Philip J Kellman and Elizabeth S Spelke. Perception of partly occluded objects in infancy. *Cognit. Psychol.*, 15(4):483–524, 1983.

[12] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[13] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *NeurIPS*, 2014.

[14] Adam Kosiorek, Hyunjik Kim, Yee Whye Teh, and Ingmar Posner. Sequential attend, infer, repeat: Generative modelling of moving objects. In *NeurIPS*, 2018.

[15] Adam Lerer, Sam Gross, and Rob Fergus. Learning physical intuition of block towers by example. In *ICML*, 2016.

[16] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

[17] Danilo J Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 2014.

[18] Danilo Jimenez Rezende, SM Eslami, Shakir Mohamed, Peter Battaglia, Max Jaderberg, and Nicolas Heess. Unsupervised learning of 3d structure from images. In *NeurIPS*, 2016.

[19] Ronan Riochet, Mario Ynocente Castro, Mathieu Bernard, Adam Lerer, Rob Fergus, Véronique Izard, and Emmanuel Dupoux. Intphys: A framework and benchmark for visual intuitive physics reasoning. *arXiv:1803.07616*, 2018.

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

[21] Kevin Smith, Lingjie Mei, Shunyu Yao, Jiajun Wu, Elizabeth Spelke, Josh Tenenbaum, and Tomer Ullman. Modeling expectation violation in intuitive physics with coarse probabilistic object representations. In *NeurIPS*, 2019.

[22] Elizabeth S Spelke and Katherine D Kinzler. Core knowledge. *Dev. Psychol.*, 10(1):89–96, 2007.

[23] Elizabeth S Spelke, Karen Breinlinger, Janet Macomber, and Kristen Jacobson. Origins of knowledge. *Psychol. Rev.*, 99(4):605, 1992.

[24] Aleksandar Stanić and Jürgen Schmidhuber. R-sqair: Relational sequential attend, infer, repeat. *arXiv:1910.05231*, 2019.

[25] Sjoerd van Steenkiste, Michael Chang, Klaus Greff, and Jürgen Schmidhuber. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. In *ICLR*, 2018.

[26] Rishi Veerapaneni, John D Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua B Tenenbaum, and Sergey Levine. Entity abstraction in visual model-based reinforcement learning. In *CoRL*, 2019.

[27] Jiajun Wu, Erika Lu, Pushmeet Kohli, Bill Freeman, and Josh Tenenbaum. Learning to see physics via visual de-animation. In *NeurIPS*, 2017.

[28] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In *NeurIPS*, 2016.