Time-Conditioned Generative Modeling of Object-Centric Representations for Video Decomposition and Prediction

Abstract

When perceiving the world from multiple viewpoints, humans have the ability to reason about the complete objects in a compositional manner even when the object is completely occluded from partial viewpoints. Meanwhile, humans can imagine the novel views after observing multiple viewpoints. The remarkable recent advance in multi-view object-centric learning leaves some problems: 1) the partially or completely occluded shape of objects can not be well reconstructed. 2) the novel viewpoint prediction depends on expensive viewpoint annotations rather than implicit view rules. This makes the agent fail to perform like humans. In this paper, we introduce a time-conditioned generative model for videos. To reconstruct the complete shape of the object accurately, we enhance the disentanglement between different latent representations: view latent representations are jointly inferred based on the Transformer and then cooperate with the sequential extension of Slot Attention to learn object-centric representations. The model also achieves the new ability: Gaussian processes are employed as priors of view latent variables for generation and novel-view prediction without viewpoint annotations. Experiments on multiple specifically designed synthetic datasets have shown that the proposed model can 1) make the video decomposition, 2) reconstruct the complete shapes of objects, and 3) make the novel viewpoint prediction without viewpoint annotations.

1 INTRODUCTION

Humans understand the multi-object world in a way that the representations of multiple objects are stored separately in terms of a compositional manner and then combined into the perceived whole [Johnson, 2010, Kahneman et al., 1992, Spelke and Kinzler, 2007]. When it comes to the multi-object scene with multiple viewpoints, humans exhibit intelligence from a higher level in multiple aspects: on the one hand, the specific object is endowed with a complete and accurate representation that captions 3D spatial structures through multi-view perception [Turnbull et al., 1997], in which case, humans have the ability to reason about the complete objects even when the object is completely occluded from partial viewpoints [Shepard and Metzler, 1971]; on the other hand, scenes corresponding to novel viewpoints can be imagined on the basis of the learned implicit rules [Schacter et al., 2012, Beaty et al., 2016]. Such compositional modeling from multiple viewpoints is the fundamental ingredient for high-level cognitive intelligence.

Unsupervised object-centric learning that is dedicated to simulating human intelligence have recently achieved remarkable advance, especially in single-view representation learning both on images [Burgess et al., 2018, Locatello et al., 2020, Emami et al., 2021, Engelcke et al., 2021] and videos [Kosiorek et al., 2018, Jiang et al., 2019, Lin et al., 2020]. Meanwhile, multi-view object-centric learning [Li et al., 2020, Chen et al., 2021, Kabra et al., 2021, Yuan et al., 2022] in synchronous progress to learn 3D object representations demonstrates a promising blueprint. However, it leaves some unresolved problems: 1) the partially or completely occluded objects from partial viewpoints cannot be reconstructed through 3D representations learned from multiple viewpoints. Although a few models can theoretically restore the occlusion, the relatively poor restoration (e.g. too much shadow, trivial object texture) is reconstructed. 2) despite using the query objective during training [Li et al., 2020], the ability for the novel viewpoint prediction depends on expensive viewpoint annotations, which provide strong location information and play significant roles in updates of object-centric representations. Conversely, the implicit rules of view representations are not fully developed to make the prediction. It is, therefore, crucial to develop a unified multi-view model to perform like humans.

Submitted to the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023). **To be used for reviewing only**.

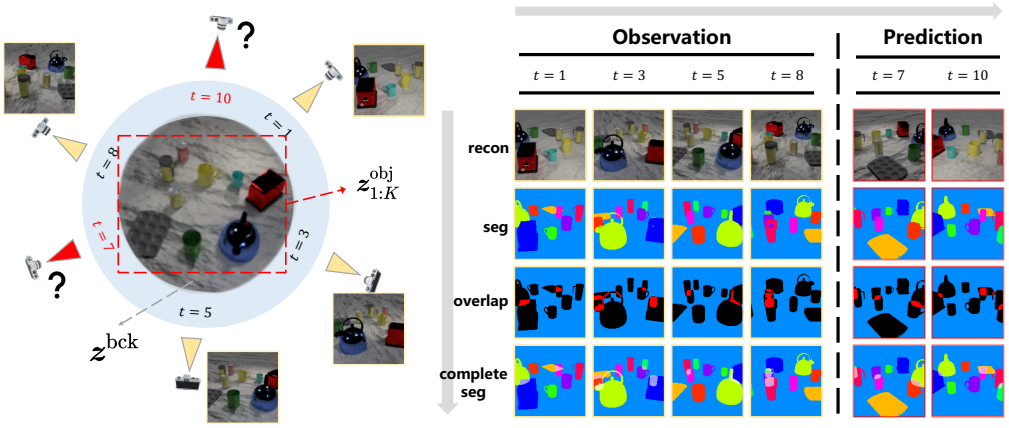


Figure 1: **Left**: multi-object video setup. Yellow and red triangles respectively represent the observed frames and predicted frames. **Right**: the description of the tasks to be solved. In our problem setting, only the observation set and times are told.

In this paper, we focus on learning object-centric representations and view representations with the potential function rules from *object-static*, *viewpoint-free* videos. The specific tasks to be solved are described in Figure 1: under the setting that only the observation set and time series are known, the model can 1) make the video decomposition based on the object-centric representations. 2) restore the complete objects even when completely occluded. 3) predict unknown viewpoints from known viewpoints and decompose the corresponding 2D images.

To address this, we build a time-conditional generative model for video decomposition and prediction. The proposed model reconstructs the complete shape of the object accurately through enhancing the disentanglement between view representations and object-centric representations: view representations are jointly inferred based on the Transformer [Vaswani et al., 2017] and then cooperate with the sequential extension of Slot Attention [Locatello et al., 2020]. Finally, the temporal mean operation maintains the 3D object-centric representations independent of view representation. Moreover, we implement the prediction from novel viewpoints without viewpoint annotations: Gaussian processes are employed as priors of view latent representations for generation and novel-view inference, facilitating the model to learn the underlying view rule. Experiments on multiple specifically designed synthetic datasets have shown that the proposed method can implement the above three tasks, moreover, outperforms the state-of-art methods in video decomposition and has competitive results on novel viewpoint predictions compared with the method that uses viewpoint annotations.

2 RELATED WORK

Object-centric learning from single view Recent breakthroughs focus on aggregating the input into multiple slots based on the attention mechanism. AIR [Eslami et al., 2016] extracts the object representations with variable numbers based on the bounding box attention [Jaderberg et al., 2015]. SQAIR [Kosiorek et al., 2018] extends AIR to videos. SPACE [Lin et al., 2019] and GMIOO [Yuan et al., 2019] both design the background modeling separately and model the occlusion from different perspectives. SCALOR [Jiang et al., 2019] implements object discovery and tracking in videos with dynamic backgrounds based on SPACE. G-SWM [Lin et al., 2020] integrates the advantages of the current model on videos and models the multimodal uncertainty. MONet [Burgess et al., 2019] uses the attention network to iteratively infer the mask and then extract representations using masked features. GENESIS [Engelcke et al., 2020] additionally models the layout of scenes based on MONet. GENESIS-V2 [Engelcke et al., 2021] infers the attention masks inspired by instance coloring applied previously in supervised instance segmentation. Slot Attention [Locatello et al., 2020] and EfficientMORL [Emami et al., 2021] randomly initialize the embedding of objects and then use a GRU to compute the feature similarities between representations and local features. ADI [Yuan et al., 2021] proposes a continual learning strategy, and has made preliminary explorations in the acquisition and utilization of knowledge.

Object-centric learning from multi view We can simplify the recent advances to two categories: models with viewpoint annotations and models without viewpoint annotations. GQN [Eslami et al., 2018] uses the viewpoint annotations to build the single-object scene. Given novel view annotations, single-object 2D images can be generated. MulMON [Li et al., 2020] models the multi-object multi-view scenes according to view annotations. The double-level iterative inference is conducted to achieve both multi-object segmentation and prediction. ROOTS [Chen et al., 2021] divides the three-dimensional space into equal-spaced grids and discovers objects in different grids. The model also con-

siders the occlusion and makes the prediction with viewpoint annotations. SIMONe [Kabra et al., 2021] and OCLOC [Yuan et al., 2022] are the recent viewpoint-free models. They learn the view representations and object-centric representations separately. The difference is that SIMONe learns representations from videos and can recompose representations to novel scenes, while OCLOC models scenes from discrete viewpoints.

3 METHOD

Our goal is to learn object-centric latent variables independent of viewpoints and structured view latent variables dependent on time t. Object-centric representations accurately describe the 3D information of objects and the background, meanwhile, view latent variables learn the correlation between different viewpoints through building the GP function of t ,by which the representations of novel viewpoints can be inferred through the function. To achieve this, multiple modules are indispensable: 1) time-conditioned generative modeling in Section 3.2 2) inference work in Section 3.3 3) a two-stage training procedure in Section 3.4. We add an additional heading to describe the different notations in Section 3.1.

3.1 DEFINITION

Suppose each *static* visual scene is composed of T consecutive viewpoints and each instance (video) is independently and identically distributed. Hereby, for simple description, the following will take a single video instance as an example. Let N and C respectively denote the number of pixels and channels in each frame $\boldsymbol{x}_t$ $(1 \leq t \leq T)$ of the video, and K denotes the maximum number of objects that appear in the visual scene. Each pixel $\boldsymbol{x}_{t,n}$ of each frame is a weighted summation of $K+1$ components at that pixel, with K describing the objects and one describing the background. $K+1$ layers of the image correspond to the $K+1$ components composed of N pixels. In the compositional modeling, layers of each frame $\boldsymbol{x}_t$ consist of pixel-wise weights $\boldsymbol{\pi}_t \in \mathbb{R}^{N\times(K+1)}$ and the expected pixel-wise RGB value $\boldsymbol{a}_t \in \mathbb{R}^{N\times(K+1)\times C}$. Both are generated by some representations (including latent variables, deterministic values, neural networks, etc.). We will express them below.

View-independent representations. we define a set of object-centric latent variables of $K+1$ entities from T viewpoints that describes the 3D visual scene, including $\boldsymbol{z}^{\text{obj}}, \boldsymbol{z}^{\text{bck}}, \boldsymbol{z}^{\text{pres}}, \boldsymbol{\nu}$.

- $\boldsymbol{z}^{\text{obj}} = \boldsymbol{z}^{\text{obj}}_{1:K}$ describes the 3D view-independent representations of objects. View-independent means the physical attributes of objects (such as shape, appearance, etc.) keep constant under different multiple viewpoints. $\boldsymbol{z}^{\text{obj}}_k$ $(1 \leq k \leq K)$ is independently and identically distributed.
- $\boldsymbol{z}^{\text{bck}}$ denotes the latent representation of the background appearance. We need not represent the shape of the background because the corresponding complete shape is 1.
- $\boldsymbol{z}^{\text{pres}} = \boldsymbol{z}^{\text{pres}}_{1:K}, \boldsymbol{\nu} = \boldsymbol{\nu}_{1:K}$ denote the latent variables that indicate the presence of objects. The advantage of using the latents is that the uncertain number of objects in different visual scenes can be added up. $z^{\text{pres}}_k (1 \leq k \leq K)$ denotes whether object k appears in a visual scene, following a Bernoulli distribution. The parameter of the distribution is controlled by latent variable ν_k that follows the conjugate prior, i.e. $z^{\text{pres}}_k \sim \text{Bernoulli}(\nu_k), \nu_k \sim \text{Beta}(\alpha/K, 1)$, where α is the hyperparameter, K denotes the object numbers.

View-dependent representations. Different from previous works [Li et al., 2020, Chen et al., 2021], we learn view representations through finding the relationship between frames, rather than directly leveraging viewpoint labels. Meanwhile, the view correlation based on temporal modeling can motivate the model to predict the novel scenes unseen given any time. The related view-dependent representations include $\boldsymbol{\lambda}, \boldsymbol{z}^{\text{view}}$.

- $\boldsymbol{\lambda} \in \mathbb{R}^{T\times D\times D_\lambda}$ represents the spatial latent variable that reflects the position characteristics of the camera under different frames, where D is the dimension of the view latent (i.e. $\boldsymbol{z}^{\text{view}}$) corresponding to the frame. D_λ is the dimension of the spatial representation that influences the meanings of each dimension in the view latent. $\boldsymbol{\lambda}_t$ potentially affects the change of viewpoints ((e.g. the distance, height, rotation of the camera) at time t. $\boldsymbol{\lambda}_{t,d}$ $(1 \leq t \leq T, 1 \leq d \leq D)$ is distributed in a linear subspace. More details can be found in 3.2.
- $\boldsymbol{z}^{\text{view}} = \boldsymbol{z}^{\text{view}}_{1:T} \in \mathbb{R}^{T\times D}$ denotes the view latent variables. Videos perform in a way that the closer the distance of two frames, the smaller the difference between the corresponding viewpoint information, and the bigger on the contrary. To build the correlation, we define $\boldsymbol{z}^{\text{view}}$ as a Gaussian process (GP) prior parameterized by the spatial latent variable $\boldsymbol{\lambda}$.

Additional Notations. In addition to the latent variables defined above, we also need some non-latent notations to generate T frames, including $\boldsymbol{s}^{\text{shp}}$, $\boldsymbol{o}$, $\boldsymbol{\pi}$, $\boldsymbol{a}$.

- $\boldsymbol{s}^{\text{shp}} \in [0,1]^{T\times N\times K}$ describes the complete shape of different objects at different time t. $\boldsymbol{s}^{\text{shp}}_{t,\cdot,k}$ ($\cdot$ represents all indexes are selected) represents the complete shape of the kth object in the 2D image corresponding to the tth frame. The range of [0, 1] guarantees the subsequent rationality of processing the occlusion. Since the complete shape of the background is a constant of 1, values of $\boldsymbol{s}^{\text{shp}}_{t,\cdot,k}$ can be computed by the neural network g_{shp} with $\boldsymbol{z}^{\text{obj}}_k$ and $\boldsymbol{z}^{\text{view}}_t$ as inputs followed by a sigmoid activation and $\boldsymbol{z}^{\text{bck}}$ need not participate in the computation.
- $\boldsymbol{o} \in \mathbb{R}^{T\times K}$ describes the occlusion order of different objects in the different frame. $o_{t,k}$ denotes the order of the kth object under the projected 2D image at the tth frame. $\boldsymbol{o}$ is obtained by the neural network g_{ord} with $\boldsymbol{z}^{\text{obj}}_k$ and $\boldsymbol{z}^{\text{view}}_t$ as inputs since the occlusion order of the same

object varies at different viewpoints. Specific usage of $\boldsymbol{o}$ can be found in 3.2.

- $\boldsymbol{\pi} \in [0,1]^{T\times N\times(K+1)}$ represents the pixel-wise weights of each layer, i.e. geometrically represents the observed shape of each object. The $\boldsymbol{\pi}$ here is different from $\boldsymbol{s}^{\text{shp}}$ in that the shape of an object may be partially observed or completely invisible due to partial or complete occlusion. $K+1$ observed shapes of the nth pixel at frame t satisfy $\sum_{k=0}^{K}\pi_{t,n,k}=1$ $(1\le t\le T, 1\le n\le N)$.
- $\boldsymbol{a} \in \mathbb{R}^{T\times N\times(K+1)\times C}$ describes the complete appearance of all entities (objects or backgrounds). $\boldsymbol{a}_{t,n,k}$ is numerically equivalent to the expected RGB value of component k at the nth pixel of the tth frame. The background appearance $\boldsymbol{a}_{t,\cdot,k}$ $(k=0)$ is achieved by the neural network $g^{\text{bck}}_{\text{apc}}$ with $\boldsymbol{z}^{\text{view}}_t$ and $\boldsymbol{z}^{\text{bck}}$ as inputs, meanwhile the kth object appearance $\boldsymbol{a}_{t,\cdot,k}$ $(1\le k\le K)$ is achieved by another neural network $g^{\text{obj}}_{\text{apc}}$ with $\boldsymbol{z}^{\text{view}}_t$ and $\boldsymbol{z}^{\text{obj}}_k$ as inputs.

Likelihood Function. After generating the observed shapes $\boldsymbol{\pi}$ and appearance $\boldsymbol{a}$ of each layer, we can use a weighted summation of each layer to reconstruct the image. Its likelihood is expressed as:

$$\log p(\boldsymbol{x}_{1:T} \mid \boldsymbol{\pi}, \boldsymbol{a}) = \sum_{t=1}^{T}\sum_{n=1}^{N}\log\mathcal{N}(\underbrace{\pi_{t,n,0}\cdot\boldsymbol{a}_{t,n,0}}_{\text{Background}} + \sum_{k=1}^{K}\underbrace{\pi_{t,n,k}\cdot\boldsymbol{a}_{t,n,k}}_{\text{Objects}}, \hat{\sigma}_x^2\boldsymbol{I}) \quad (1)$$

where $\hat{\sigma}_x$ is the hyperparameter. The style of the likelihood function is similar to Slot Attention [Locatello et al., 2020] in order to improve the reconstruction.

3.2 GENERATIVE MODELING

Let $\boldsymbol{x}_S = \boldsymbol{x}_{1:T}$ be T images in a visual scene and $\boldsymbol{t}_{\mathcal{S}}$ be the all timesteps. Each scene can be (arbitrarily) divided into observation frames $\boldsymbol{x}_{\mathcal{T}}$ and prediction frames $\boldsymbol{x}_{\mathcal{Q}}$, where $\boldsymbol{x}_{\mathcal{T}} + \boldsymbol{x}_{\mathcal{Q}} = \boldsymbol{x}_{\mathcal{S}}$ and $\boldsymbol{x}_{\mathcal{T}}$ $(\boldsymbol{x}_{\mathcal{Q}})$ keeps a chronological order, e.g. $\boldsymbol{x}_{\mathcal{T}} = \big(\boldsymbol{x}_1, \boldsymbol{x}_3, \boldsymbol{x}_7, \boldsymbol{x}_9\big)$ is sorted according to the time. $\boldsymbol{t}_{\mathcal{S}}$ can be correspondingly divided into $\boldsymbol{t}_{\mathcal{T}}$ and $\boldsymbol{t}_{\mathcal{Q}}$.The generative model conditional on time $\boldsymbol{t}_{\mathcal{S}}$ can be expressed as (a intuitive diagram can see in the supplementary materials):

$$\boldsymbol{\lambda}_{t,d} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{w}_t, \sigma_w^2\boldsymbol{I}) \quad (2)$$

$$\kappa^d_{\boldsymbol{\eta}}(\boldsymbol{\lambda}_i, \boldsymbol{\lambda}_j) = l^2\exp\Big(\frac{\|g^d_{\boldsymbol{\eta}}(\boldsymbol{\lambda}_i) - g^d_{\boldsymbol{\eta}}(\boldsymbol{\lambda}_j)\|_2^2}{2\sigma^2}\Big) \quad (3)$$

$$\boldsymbol{z}^{\text{obj}}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}); \qquad \boldsymbol{z}^{\text{bck}} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \quad (4)$$

$$\boldsymbol{K}^d_{\boldsymbol{\eta}} = \begin{bmatrix} \kappa^d_{\boldsymbol{\eta}}(\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_1) & \cdots & \kappa^d_{\boldsymbol{\eta}}(\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_T) \\ \vdots & \ddots & \vdots \\ \kappa^d_{\boldsymbol{\eta}}(\boldsymbol{\lambda}_T, \boldsymbol{\lambda}_1) & \cdots & \kappa^d_{\boldsymbol{\eta}}(\boldsymbol{\lambda}_T, \boldsymbol{\lambda}_T) \end{bmatrix} \quad (5)$$

$$\boldsymbol{z}^{\text{view}}_{1:T,d} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}^d_{\boldsymbol{\eta}}) \quad (6)$$

$$\boldsymbol{z}^{\text{view}}_{1:T} = \text{concat}(\boldsymbol{z}^{\text{view}}_{\cdot,1}, ..., \boldsymbol{z}^{\text{view}}_{\cdot,D}, \text{axis} = \text{“view dim”}) \quad (7)$$

$$\nu_k \sim \text{Beta}(\alpha/K, 1); \qquad z^{\text{pres}}_k \sim \text{Bernoulli}(\nu_k) \quad (8)$$

$$s^{\text{shp}}_{t,n,k} = \text{Sigmoid}(g_{\text{shp}}(\boldsymbol{z}^{\text{obj}}_k, \boldsymbol{z}^{\text{view}}_t)_n) \quad (9)$$

$$o_{t,k} = g_{\text{ord}}(\boldsymbol{z}^{\text{obj}}_k, \boldsymbol{z}^{\text{view}}_t) \quad (10)$$

$$\pi_{t,n,k} = \begin{cases} \prod_{k'=1}^{K}(1 - z^{\text{pres}}_{k'}s^{\text{shp}}_{t,n,k'}), & k=0 \\ \frac{(1-\pi_{t,n,0})(1-z^{\text{pres}}_k s^{\text{shp}}_{t,n,k}o_{t,k})}{\sum_{k'=1}^{K}1-z^{\text{pres}}_{k'}s^{\text{shp}}_{t,n,k'}o_{t,k'}}, & k\ge 1 \end{cases} \quad (11)$$

$$\boldsymbol{a}_{t,n,k} = \begin{cases} g^{\text{bck}}_{\text{apc}}(\boldsymbol{z}^{\text{view}}_t, \boldsymbol{z}^{\text{bck}})_n, & k=0 \\ g^{\text{obj}}_{\text{apc}}(\boldsymbol{z}^{\text{view}}_t, \boldsymbol{z}^{\text{apc}}_k)_n, & k\ge 1 \end{cases} \quad (12)$$

$$\boldsymbol{x}_{t,n} \sim \mathcal{N}(\sum\nolimits_{k=0}^{K}\pi_{t,n,k}\cdot\boldsymbol{a}_{t,n,k}, \sigma_x^2\boldsymbol{I}) \quad (13)$$

Here all subscript ranges $(1\le t\le T, 1\le d\le D, 1\le k\le K, 1\le n\le N)$ are omitted for simplicity. In the above mathematical expression, $\boldsymbol{w}_t$ is the time encoding w.r.t. t, i.e. $\boldsymbol{w}_t = \text{TimestepEncoding}(t)$. Choices to encode the timestep are diverse, e.g. $\boldsymbol{w}_t = \big[\cos t, \sin t\big]$. $\boldsymbol{\lambda}_{t,d}$ follows a linear distribution with an affine matrix $\boldsymbol{A}$ (learned or fixed) and σ_w is the hyperparameter. $\kappa^d_{\boldsymbol{\eta}}$ is the kernel function corresponding to the dth dimension of $\boldsymbol{z}^{\text{view}}$ composed of a neural network $g^d_{\boldsymbol{\eta}}$ and RBF kernel parameterized with $\boldsymbol{\eta}$, l and σ ([Wilson et al., 2016]). Each dimension of the view latent variable is generated by a different GP and finally concatenate together (see Eq 6). Eq 11 processes the occlusion through sorting the hierarchical relationship to get the observed shape. Let $\boldsymbol{\Omega} = \big\{\boldsymbol{z}^{\text{bck}}, \boldsymbol{z}^{\text{obj}}, \boldsymbol{z}^{\text{pres}}, \boldsymbol{\nu}, \boldsymbol{\lambda}, \boldsymbol{z}^{\text{view}}\big\}$ represents the collection of all latent variables, the joint conditional probability of $\boldsymbol{x}_{\mathcal{S}}$ and $\boldsymbol{\Omega}$ can be written as:

$$p(\boldsymbol{x}_{\mathcal{S}}, \boldsymbol{\Omega} \mid \boldsymbol{t}_{\mathcal{S}}) = \prod\nolimits_{t=1}^{T}\prod\nolimits_{n=1}^{N}p(\boldsymbol{x}_{t,n} \mid \boldsymbol{\Omega})p(\boldsymbol{z}^{\text{bck}}) \prod\nolimits_{k=1}^{K}p(\boldsymbol{z}^{\text{obj}}_k)p(\nu_k)p(z^{\text{pres}}_k \mid \nu_k) \prod\nolimits_{d=1}^{D}\prod\nolimits_{t=1}^{T}p(\boldsymbol{z}^{\text{view}}_{\mathcal{S},d} \mid \boldsymbol{\lambda}_{\mathcal{S},d})p(\boldsymbol{\lambda}_{t,d} \mid \boldsymbol{t}_{\mathcal{S}}) \quad (14)$$

3.3 INFERENCE

Since we cannot compute the likelihood by integrating latent variables $\boldsymbol{\Omega}$, amortized variational inference based on inference networks is employed to approximate the posterior of latent variables. In our problem setting, only part of the frame collection $\boldsymbol{x}_{\mathcal{S}}$ for each scene is observed. This means $\boldsymbol{z}^{\text{view}}_{\mathcal{T}}$ and $\boldsymbol{\lambda}_{\mathcal{T}}$ that belong to $\boldsymbol{x}_{\mathcal{T}}$ can be inferred directly from inference networks, while the posteriors of $\boldsymbol{z}^{\text{view}}_{\mathcal{Q}}$ and $\boldsymbol{\lambda}_{\mathcal{Q}}$ that belong to $\boldsymbol{x}_{\mathcal{Q}}$ are hard to compute. We use the least square method to approximate the posterior of $\boldsymbol{\lambda}_{\mathcal{Q}}$ and then explicitly compute the posterior of $\boldsymbol{z}^{\text{view}}_{\mathcal{Q}}$ based on the properties of the GP prior. To simplify the description, let the parameters in the inference network be $\boldsymbol{\phi}$ and the parameters in the learnable kernels in GP be $\boldsymbol{\eta}$. The variational posterior $q_{\boldsymbol{\phi},\boldsymbol{\eta}}(\boldsymbol{\Omega} \mid \boldsymbol{x}_{\mathcal{T}}, \boldsymbol{t}_{\mathcal{T}})$ conditional on the observed set $\boldsymbol{x}_{\mathcal{T}}$ and $\boldsymbol{t}_{\mathcal{T}}$ can be written as:

$$q_{\boldsymbol{\phi},\boldsymbol{\eta}}(\boldsymbol{\Omega} \mid \boldsymbol{x}_{\mathcal{T}}, \boldsymbol{t}_S) = q_{\boldsymbol{\phi}}(\boldsymbol{z}^{\text{bck}} \mid \boldsymbol{x}_{\mathcal{T}})q_{\boldsymbol{\phi}}(\boldsymbol{z}^{\text{view}}_{\mathcal{T}} \mid \boldsymbol{x}_{\mathcal{T}}, \boldsymbol{t}_{\mathcal{T}})$$

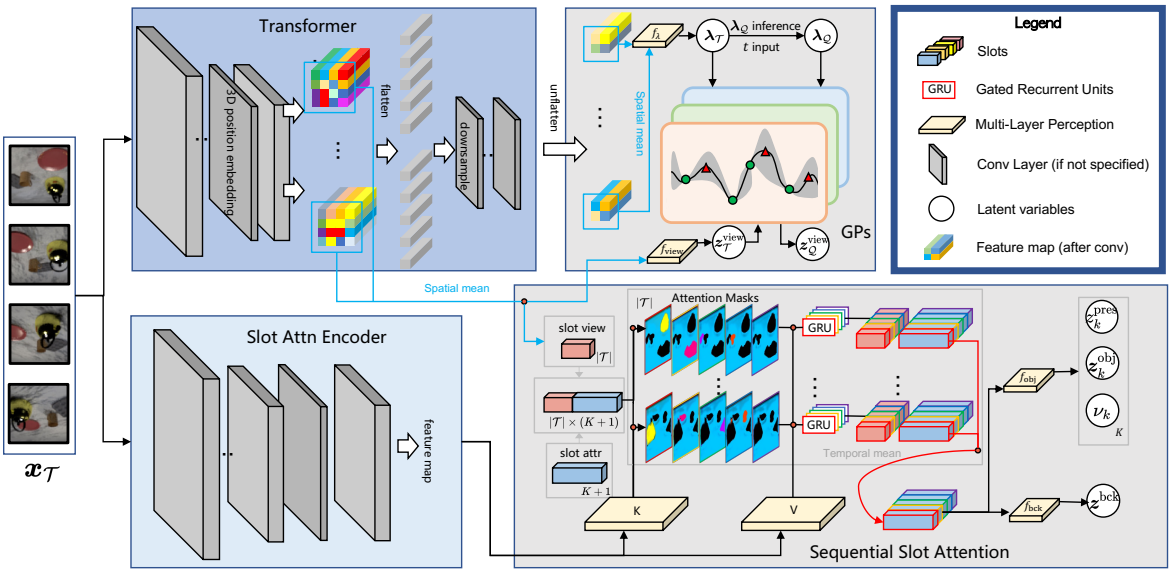


Figure 2: The inference process of the model. **Left**: observation inputs; **Middle**: the object-centric representation encoder and view representation encoder; **Right**: GP prediction module and sequential Slot Attention.

$$q_\phi(\boldsymbol{\lambda}_\mathcal{T} \mid \boldsymbol{x}_\mathcal{T}, \boldsymbol{t}_\mathcal{T}) q_\phi(\boldsymbol{\lambda}_\mathcal{Q} \mid \boldsymbol{\lambda}_\mathcal{T}, \boldsymbol{t}_\mathcal{Q}) \prod_{d=1}^{D} q_\boldsymbol{\eta}(\boldsymbol{z}_{\mathcal{Q},d}^{\text{view}} \mid \boldsymbol{z}_{\mathcal{T},d}^{\text{view}}, \boldsymbol{\lambda}_\mathcal{S})$$
$$\prod_{k=1}^{K} q_\phi(\boldsymbol{z}_k^{\text{obj}} \mid \boldsymbol{x}_\mathcal{T}) q_\phi(\nu_k \mid \boldsymbol{x}_\mathcal{T}) q_\phi(z_k^{\text{pres}} \mid \boldsymbol{x}_\mathcal{T}) \tag{15}$$

We will respectively describe the inference procedure of object-centric latent variables (corresponding to 3.3.3) and the view-dependent latent variables, in which the inference of view-dependent latent variables can be further subdivided into the inference of observation (corresponding to 3.3.1) and prediction (corresponding to 3.3.2). The specific overview of inference can be found in Figure 2 and the mathematical form is described in the supplementary material.

3.3.1 Inference of observed view-dependent latents

The specific inference of each distribution will be briefly described in conjunction with Figure 2 and algorithm 2 in Appendix B. The posteriors of the view latent variable $\boldsymbol{z}_t^{\text{view}}$ $(t \in \mathcal{T})$ and spatial latent variable $\boldsymbol{\lambda}_{t,d}$ $(t \in \mathcal{T}, 1 \leq d \leq D)$ are described as:

$$q(\boldsymbol{z}_t^{\text{view}} \mid \boldsymbol{x}_\mathcal{T}) = \mathcal{N}(\boldsymbol{z}_t^{\text{view}} \mid \boldsymbol{\mu}_t^{\text{view}}, \text{diag}(\boldsymbol{\sigma}_t^{\text{view}})^2) \tag{16}$$
$$q(\boldsymbol{\lambda}_{t,d} \mid \boldsymbol{x}_\mathcal{T}, \boldsymbol{t}_\mathcal{T}) = \mathcal{N}(\boldsymbol{\lambda}_{t,d} \mid \boldsymbol{\mu}_{t,d}(\boldsymbol{x}_\mathcal{T}, \boldsymbol{w}_t), \sigma_\boldsymbol{w}^2 \boldsymbol{I}) \tag{17}$$

$\boldsymbol{z}^{\text{view}}$ is inferred based on the Transformer [Vaswani et al., 2017], where $\boldsymbol{y}_\mathcal{T}^{\text{feat}}$ extracted by the neural network f_{feat} (with $\boldsymbol{x}_\mathcal{T}$ as inputs) will enter into the Multi-SelfAttention module after given a 3D position embedding. The Transformer module learns the viewpoint information with frame correlation. The $|\mathcal{T}| \times L \times C$ feature map extracted will be averaged over dimension L to obtain $\boldsymbol{y}_t^{\text{view}}$ $(t \in \mathcal{T})$, where L is product of length (H) and width (W) corresponding to extracted feature map. $\boldsymbol{\mu}_t^{\text{view}}$ and $\boldsymbol{\sigma}_t^{\text{view}}$ of $\boldsymbol{z}_t^{\text{view}}$ are parameterized by the neural network f_{view}.

$\boldsymbol{\mu}$ of the spatial latent variable $\boldsymbol{\lambda}_\mathcal{Q}$ is parameterized by the neural network f_λ with $y_\mathcal{T}^\lambda$ and $\boldsymbol{w}_\mathcal{T}$ as inputs, where $\boldsymbol{y}_\mathcal{T}^\lambda$ represents the more abstract concepts after the Downsample and Multi-SelfAttention operation (Figure 2) and $\boldsymbol{w}_\mathcal{T}$ is the timestep encoding of $\mathcal{T}$. The variance is equal to $\boldsymbol{\sigma}^2$.

3.3.2 Inference of predicted view-dependent latents

Inference of latent variables related to predicted viewpoints is difficult because no information is provided by $\boldsymbol{x}_\mathcal{Q}$. Therefore, the predicted view-dependent latents needs to be inferred from the observed viewpoints. We will respectively describe the inference method of $\boldsymbol{\lambda}_\mathcal{Q}$ and $\boldsymbol{z}_\mathcal{Q}^{\text{view}}$.

inference of $\boldsymbol{\lambda}_\mathcal{Q}$ According to the definition of prior $\boldsymbol{\lambda}$ in Eq 2, each $\boldsymbol{\mu}_{t,d}$ of posterior $q(\boldsymbol{\lambda}_\mathcal{T} \mid \boldsymbol{x}_\mathcal{T})$ can be approximated to satisfy a linear function w.r.t. $\boldsymbol{w}_t$, i.e. $\boldsymbol{\mu}_{t,d} = \boldsymbol{A}_d \boldsymbol{w}_t$, $\boldsymbol{A}_d \in \mathbb{R}^{D_\lambda \times |\boldsymbol{w}|}$, Based on the Least Square Error method, the optimal $\hat{\boldsymbol{A}}_d^*$ $(1 \leq d \leq D)$ in the linear function set and the posterior of $\boldsymbol{\lambda}_\mathcal{Q}$ can be desrcibed as:

$$q(\boldsymbol{\lambda}_{t,d} \mid \boldsymbol{x}_\mathcal{T}, \boldsymbol{t}_\mathcal{S}) = \mathcal{N}(\hat{\boldsymbol{A}}_d^* \boldsymbol{w}_t, \sigma_\boldsymbol{w}^2 \boldsymbol{I}) \tag{18}$$
$$\hat{\boldsymbol{A}}_d^* = \boldsymbol{\Phi}_d^\top \boldsymbol{W}_\mathcal{T} (\boldsymbol{W}_\mathcal{T}^\top \boldsymbol{W}_\mathcal{T})^{-1} \tag{19}$$

where $\boldsymbol{W}_\mathcal{T} = \left[\boldsymbol{w}_1, ..., \boldsymbol{w}_{|\mathcal{T}|}\right]^\top \in \mathbb{R}^{|\mathcal{T}| \times |\boldsymbol{w}|}$, $\boldsymbol{\Phi}_d = \left[\boldsymbol{\mu}_{1,d}, ..., \boldsymbol{\mu}_{|\mathcal{T}|,d}\right]^\top \in \mathbb{R}^{|\mathcal{T}| \times D_\lambda}$.

inference of $\boldsymbol{z}_\mathcal{Q}^{\text{view}}$ $q(\boldsymbol{z}_\mathcal{Q}^{\text{view}} \mid \boldsymbol{z}_\mathcal{T}^{\text{view}}, \boldsymbol{\lambda}_\mathcal{S})$ follows the same calculation manner as the prediction with GPs, i.e

$$q(\boldsymbol{z}_\mathcal{Q}^{\text{view}} \mid \boldsymbol{z}_\mathcal{T}^{\text{view}}, \boldsymbol{\lambda}_\mathcal{S}) = \prod_{d=1}^{D} p_\boldsymbol{\eta}(\boldsymbol{z}_{\mathcal{Q},d}^{\text{view}} \mid \boldsymbol{z}_{\mathcal{T},d}^{\text{view}}, \boldsymbol{\lambda}_{\mathcal{S},d}) \tag{20}$$

where $p_{\boldsymbol{\eta}}(\boldsymbol{z}^{\text{view}}_{\mathcal{Q},d} \mid \cdot)$ satisfies the multivariate Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_{\mathcal{Q},d}, \boldsymbol{\Sigma}_{\mathcal{Q},d})$, and the parameters $\boldsymbol{\mu}_{\mathcal{Q},d}$ and $\boldsymbol{\Sigma}_{\mathcal{Q},d}$ of $\boldsymbol{z}^{\text{view}}_{\mathcal{Q},d}$ are analytical functions of $\boldsymbol{\lambda}_{\mathcal{S},d}$, $\boldsymbol{z}^{\text{view}}_{\mathcal{T},d}$ and $\boldsymbol{\eta}$.

3.3.3 Inference of view-independent latents

The posteriors of the object-centric latent representations, i.e. $\{\boldsymbol{z}^{\text{bck}}, \boldsymbol{z}^{\text{obj}}, \boldsymbol{z}^{\text{pres}}, \boldsymbol{\nu}\}$ in Eq 15 are described as:

$$q_{\boldsymbol{\phi}}(\boldsymbol{z}^{\text{bck}} \mid \boldsymbol{x}_{\mathcal{T}}) = \mathcal{N}(\boldsymbol{z}^{\text{bck}} \mid \boldsymbol{\mu}^{\text{bck}}, \text{diag}(\boldsymbol{\sigma}^{\text{bck}})^2) \tag{21}$$

$$q_{\boldsymbol{\phi}}(\boldsymbol{z}^{\text{obj}}_k \mid \boldsymbol{x}_{\mathcal{T}}) = \mathcal{N}(\boldsymbol{z}^{\text{obj}}_k \mid \boldsymbol{\mu}^{\text{obj}}_k, \text{diag}(\boldsymbol{\sigma}^{\text{obj}}_k)^2) \tag{22}$$

$$q_{\boldsymbol{\phi}}(\nu_k \mid \boldsymbol{x}_{\mathcal{T}}) = \text{Beta}(\nu_k \mid \tau_{k,1}, \tau_{k,2}) \tag{23}$$

$$q_{\boldsymbol{\phi}}(z^{\text{pres}}_k \mid \boldsymbol{x}_{\mathcal{T}}) = \text{Bernoulli}(z^{\text{pres}}_k \mid \kappa_k) \tag{24}$$

where the default range of k is $1 \leq k \leq K$. All the parameters of distributions will pass through sequential extension of Slot Attention [Locatello et al., 2020] and we will briefly describe the overview in Figure 2.

The model maintains $K+1$ attribute slots $\boldsymbol{y}^{\text{attr}} = [\boldsymbol{y}^{\text{bck}}, \boldsymbol{y}^{\text{obj}}_1, ..., \boldsymbol{y}^{\text{obj}}_K]$, where K object slots $\boldsymbol{y}^{\text{obj}}_k \in \mathbb{R}^{D_s}$ and 1 background slot $\boldsymbol{y}^{\text{bck}} \in \mathbb{R}^{D_s}$. Different from Slot Attention [Locatello et al., 2020], two types of initialization are employed for the foreground and background. Then $\boldsymbol{y}^{\text{attr}}_k$ will combine with $\boldsymbol{y}^{\text{view}}_t \in \mathbb{R}^{D_v}$ $(t \in \mathcal{T})$ extracted from the Transformer (see 3.3.1) to be $|\mathcal{T}| \times (K+1)$ slots $\boldsymbol{y}^{\text{full}}_{t,k} \in \mathbb{R}^{D_f}$ with viewpoint information, where $D_f = D_s + D_v$. We use another encoder with $\boldsymbol{x}_{\mathcal{T}}$ as inputs to extract $\boldsymbol{y}^{\text{sa}}_{\mathcal{T}}$. Then the cross attention is emplyed to decompose the inputs to multiple slots based on softmax normalization of similarity calculation. Different from the standard Slot Attention, we execute the temporal mean over the updated attribute part of $\boldsymbol{y}^{\text{full}}_{t,k}$ during each iteration:

$$\boldsymbol{a}_t = \underset{K+1}{\text{softmax}}\Big(\frac{k(\boldsymbol{y}^{\text{sa}}_t) \cdot q(\boldsymbol{y}^{\text{full}}_{t,1:K+1})^\top}{\sqrt{D_f}}\Big) \in \mathbb{R}^{N \times K} \tag{25}$$

$$\boldsymbol{u}_t = \sum_N \underset{N}{\text{softmax}}\Big(\log \boldsymbol{a}_{t,n}\Big) \cdot v(\boldsymbol{y}^{\text{sa}}_{t,n}) \in \mathbb{R}^{K \times D_f} \tag{26}$$

$$\hat{\boldsymbol{y}}^{\text{full}}_{t,k} = \text{GRU}(\boldsymbol{y}^{\text{full}}_{t,k}, \boldsymbol{u}_{t,k}), \quad [\hat{\boldsymbol{y}}^{\text{view}}_{t,k}, \hat{\boldsymbol{y}}^{\text{attr}}_{t,k}] \overset{\text{split}}{\leftarrow} \hat{\boldsymbol{y}}^{\text{full}}_{t,k} \tag{27}$$

$$\boldsymbol{y}^{\text{attr}}_k = \text{mean}_{|\mathcal{T}|}\Big(\hat{\boldsymbol{y}}^{\text{attr}}_{1:|\mathcal{T}|,k}\Big) \tag{28}$$

where k,q and v respectively represent the key, query and value. The procedure maintains the permutation invariance w.r.t. the input order. $\boldsymbol{\mu}^{\text{bck}}, \boldsymbol{\sigma}^{\text{bck}}$ will be respectively parameterized by the neural network f_{bck} and $\boldsymbol{\mu}^{\text{obj}}_k, \boldsymbol{\sigma}^{\text{obj}}_k, \tau_{k,1}, \tau_{k,2}, \kappa_k$ are parameterized by the shared neural network f_{obj}.

3.4 TRAINING

We found optimizing the evidence lower bound (ELBO) corresponding to observed and predicted frames is unstable. To solve this problem, a two-stage training procedure is used. Hereby, let the latent variables $\boldsymbol{\Omega} = \{\boldsymbol{\Omega}_{\mathcal{T}}, \boldsymbol{\Omega}_{\mathcal{Q}}\}$, where view-independent latent variables share between both, i.e. $\boldsymbol{\Omega}_{\mathcal{T}} = \{\boldsymbol{z}^{\text{bck}}, \boldsymbol{z}^{\text{obj}}, \boldsymbol{z}^{\text{pres}}, \boldsymbol{\nu}, \boldsymbol{\lambda}_{\mathcal{T}}, \boldsymbol{z}^{\text{view}}_{\mathcal{T}}\}$, $\boldsymbol{\Omega}_{\mathcal{Q}} = \{\boldsymbol{z}^{\text{bck}}, \boldsymbol{z}^{\text{obj}}, \boldsymbol{z}^{\text{pres}}, \boldsymbol{\nu}, \boldsymbol{\lambda}_{\mathcal{Q}}, \boldsymbol{z}^{\text{view}}_{\mathcal{Q}}\}$. Two-stage losses are described as:

$$\begin{aligned}\mathcal{L}_1 = &-\mathbb{E}_{q_{\boldsymbol{\phi},\boldsymbol{\eta}}(\boldsymbol{\Omega}_{\mathcal{T}}|\boldsymbol{x}_{\mathcal{T}})}\big[\log p_{\boldsymbol{\theta},\boldsymbol{\eta}}(\boldsymbol{x}_{\mathcal{T}} \mid \boldsymbol{\Omega}_{\mathcal{T}})\big] \\ &+ D_{KL}\Big(q_{\boldsymbol{\phi},\boldsymbol{\eta}}(\boldsymbol{\Omega}_{\mathcal{T}} \mid \boldsymbol{x}_{\mathcal{T}}) \| p_{\boldsymbol{\theta},\boldsymbol{\eta}}(\boldsymbol{\Omega}_{\mathcal{T}})\Big)\end{aligned} \tag{29}$$

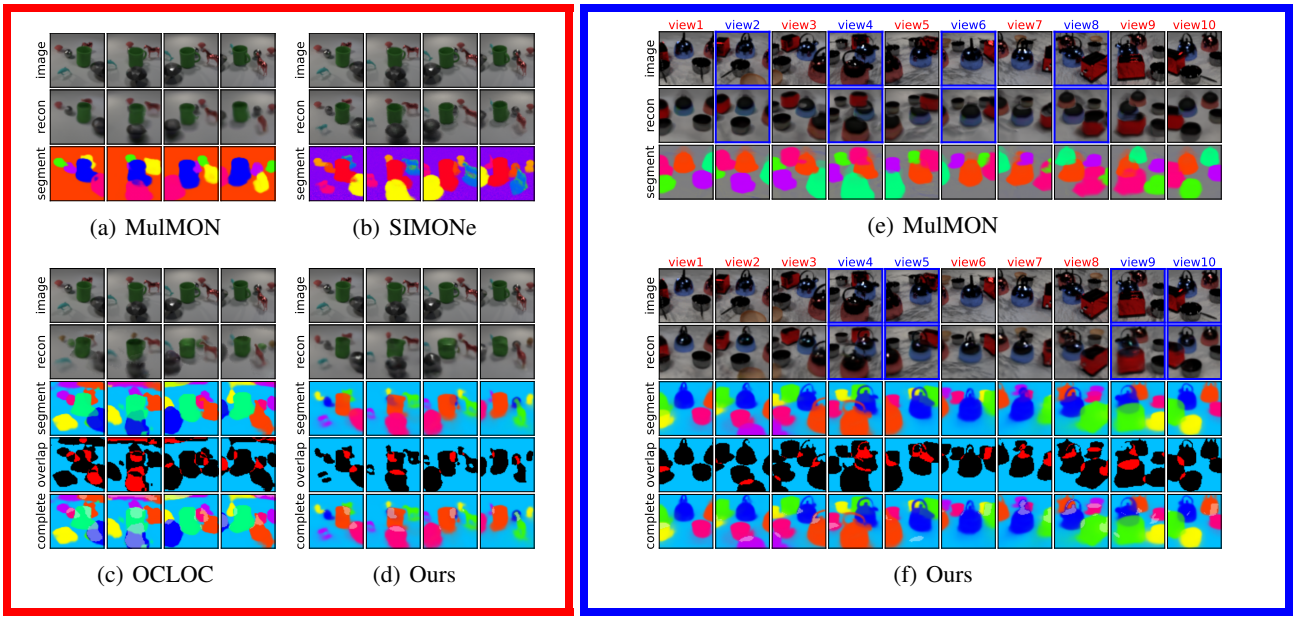
$$\begin{aligned}\mathcal{L}_2 = &-\frac{1}{|\mathcal{T}|}\mathbb{E}_{q_{\boldsymbol{\phi},\boldsymbol{\eta}}(\boldsymbol{\Omega}_{\mathcal{T}}|\boldsymbol{x}_{\mathcal{T}},\boldsymbol{t}_{\mathcal{T}})}\big[\log p_{\boldsymbol{\phi},\boldsymbol{\eta}}(\boldsymbol{x}_{\mathcal{T}} \mid \boldsymbol{\Omega}_{\mathcal{T}})\big] \\ &-\frac{1}{|\mathcal{Q}|}\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\Omega}_{\mathcal{T}}|\cdot)q_{\boldsymbol{\phi},\boldsymbol{\eta}}(\boldsymbol{\Omega}_{\mathcal{Q}}|\boldsymbol{\Omega}_{\mathcal{T}},\boldsymbol{t}_{\mathcal{Q}})}\big[\log p_{\boldsymbol{\theta},\boldsymbol{\eta}}(\boldsymbol{x}_{\mathcal{Q}} \mid \boldsymbol{\Omega}_{\mathcal{Q}})\big] \\ &+ \beta D_{KL}\Big(q_{\boldsymbol{\phi},\boldsymbol{\eta}}(\boldsymbol{\Omega}_{\mathcal{S}} \mid \boldsymbol{x}_{\mathcal{T}}, \boldsymbol{t}_{\mathcal{S}}) \| p_{\boldsymbol{\theta},\boldsymbol{\eta}}(\boldsymbol{\Omega}_{\mathcal{S}} \mid \boldsymbol{t}_{\mathcal{S}})\Big)\end{aligned} \tag{30}$$

where $\mathcal{L}_1$ is a standard ELBO of $\boldsymbol{x}_{\mathcal{T}}$ in order to learn multi-object representations from multiple views and does not use $\boldsymbol{t}_{\mathcal{S}}$ while $\mathcal{L}_2$ adopts the curriculum learning to learn the function of view latents w.r.t. $\boldsymbol{t}_{\mathcal{S}}$. let $\mathcal{S}'$ denote the subset of $\mathcal{S}$ and $|\mathcal{S}'|$ will gradually increase during training. $\mathcal{S}'$ will be randomly divided into $\mathcal{T}$ and $\mathcal{Q}$, where $|\mathcal{Q}| \sim U(1, K)$ (K dynamically increases during training). The model averages the observed and predicted losses to balance the differene between two losses. $\beta \geq 1$ follows [Burgess et al., 2018]. Note that the reconstruction performance of $\mathcal{L}_2$ is worse than that of stage 1, but it can perform well on the prediction task.

4 EXPERIMENTS

We design a bunch of experiments to investigate 1) how our proposed model compares to existing multi-view models among the observation in terms of video decomposition, 2) whether our proposed model disentangles the scene into object-centric representations and view representations, 3) how accurately our proposed model handles the occlusion problem compare with existing methods, 4) how accurately our model makes the prediction only depending on t.

To do so, we compare the performance against three baseline models, MulMON [Li et al., 2020] with viewpoint annotations, viewpoint-free models SIMONe [Kabra et al., 2021] and OCLOC [Yuan et al., 2022]. We design four synthetic multi-view video datasets called CLEVR-SIMPLE, CLEVR-COMPLE, SHOP-SIMPLE, SHOP-COMPLEX. multi-view CLEVR [Johnson et al., 2017] and SHOP [Nazarczuk and Mikolajczyk, 2020] are modified according to the official code. SHOP is harder than CLEVR in terms of the object texture and COMPLEX version is harder than SIMPLE because of more types of objects and backgrounds. We train the model with a two-stage procedure. Stage 1 can reconstruct the observation set without supervision while Stage 2 can predict unobserved set only with time supervision. We train all the datasets using an Adam optimizer with a learning rate 4e-4 for $300k$ gradient steps. The increment



(a) MulMON (b) SIMONe (c) OCLOC (d) Ours (e) MulMON (f) Ours

Figure 3: **Left**: Representative qualitative results of the **observation** set in CLEVER-COMPLEX datasets. 4 consecutive frames are demonstrated. **Right**: Representative qualitative results of the **prediction** set in SHOP-SIMPLE datasets. The images (*G.T.*) framed in blue are untold, and recons in blue are predicted images.

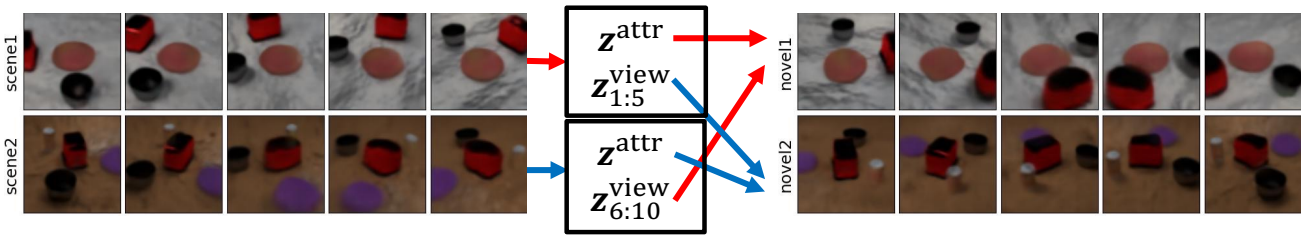


Figure 4: generate novel 2D images from novel views through recomposing view representations and object-centric representations. Here shows the results of SHOP-COMPLEX.

of curriculum learning is 2.

Video Decomposition. Since the model maintains the view-independent object-centric representations with 3D structure, video decomposition is crucial to measuring the completeness and accuracy of representations. Figure 3 (Left) demonstrates the visualization results of CLEVR-COMPLEX. Our model can accurately represent objects with complex shapes from multiple viewpoints. Furthermore, the model can build crisp segregation between the foregrounds and the background. Moreover, the proposed model tends to treat shadows as parts of objects (e.g., the horse in Figure 3 (Left: (d)), it's reasonable for shadows to blend with objects according to the lighting. Surprisingly, the shadow area is noticeably smaller than other models. Table 1 (a) demonstrates the segmentation performance in terms of foregrounds. ARI-O measures how accurately videos are decomposed into different objects. We find that except for CLEVR-SIMPLE, our proposed model outperforms other models, especially in SHOP datasets, probably because the 3D representations integrity of objects helps reconstruct better masks. SIMONe and OCLOC fail to capture the objects in SHOP-COMPLEX. A possible reason is that the background is highly coincident with objects in SHOP-COMPLEX, so that the model cannot represent the background separately during the inference. Although OCLOC separately models the background, sampling from permutation-equivalent slots may affect the extraction of the background representation.

Video Recomposition. An intriguing experiment is to generate 2D images from novel views through cross-combining view representations and attribute presentations of objects (including z^{bck} and z^{obj}). Here is the description how we execute the recomposition in Figure 4: First, we randomly choose two videos (10 frames); then, take the first 5 frames and the last 5 frames respectively and encode them into the corresponding view representations and object-centric representations; finally, we concatenate two representations from different videos to decode the novel viewpoints. Figure 4 demonstrates that representations from different scenes can be well coupled, and the model can generate novel views through recomposition. This indirectly proves the correctness of decomposing the two representations.

Table 1: Quantitative comparisons of MulMON, SIMONe and Our proposed model. except for MSE in (d), all results are recorded mean ± standard error (5 seeds). S means SIMPLE and C means COMPLEX.

(a) Observed ARI-O segmentation performance

Model	CLEVR-S	CLEVR-C	SHOP-S	SHOP-C
	ARI-O↑	ARI-O↑	ARI-O↑	ARI-O↑
MulMON (cond)	**96.4** ± 0.1	92.9 ± 0.2	88.3 ± 0.6	87.1 ± 0.2
SIMONe	91.0 ± 0.0	91.4 ± 0.0	55.3 ± 0.0	33.5 ± 0.0
OCLOC	92.7 ± 0.8	82.7 ± 0.8	91.3 ± 0.4	29.3 ± 0.5
Ours	95.9 ± 0.3	**94.1** ± 0.3	**95.8** ± 0.1	**94.9** ± 0.4

(b) Observed IoU,OOA occlusion performance

Model	IoU↑		OOA↑	
	OCLOC	Ours	OCLOC	Ours
CLEVR-S	45.6 ± 0.2	**59.5** ± 0.5	93.6 ± 1.2	**95.3** ± 1.1
CLEVR-C	35.1 ± 0.2	**50.9** ± 0.4	89.1 ± 1.2	**93.0** ± 0.8
SHOP-S	61.9 ± 0.6	**65.9** ± 0.1	72.8 ± 1.4	**78.9** ± 0.4
SHOP-C	21.5 ± 0.3	**66.2** ± 0.6	57.9 ± 1.9	**81.8** ± 1.3

(c) Predicted ARI-O segmentation performance

Model		CLEVR-S	CLEVR-C	SHOP-S	SHOP-C
		ARI-O↑	ARI-O↑	ARI-O↑	ARI-O↑
mode1	MulMON	**96.2** ± 0.1	91.5 ± 0.3	88.3 ± 0.5	86.9 ± 0.7
	Ours	95.5 ± 0.5	**95.5** ± 0.9	**96.0** ± 0.3	**92.9** ± 0.4
mode2	MulMON	**96.9** ± 0.2	94.5 ± 0.2	87.1 ± 0.6	86.0 ± 0.6
	Ours	95.1 ± 0.5	**95.0** ± 0.6	**95.5** ± 0.1	**93.8** ± 0.8

(d) Predicted MSE reconstruction performance

Model		CLEVR-S	CLEVR-C	SHOP-S	SHOP-C
		MSE↓	MSE↓	MSE↓	MSE↓
mode1	MulMON	**0.0014**	**0.0020**	0.0049	0.0038
	Ours	0.0018	0.0021	**0.0034**	**0.0036**
mode2	MulMON	**0.0014**	**0.0020**	0.0050	0.0038
	Ours	0.0017	0.0024	**0.0035**	**0.0038**

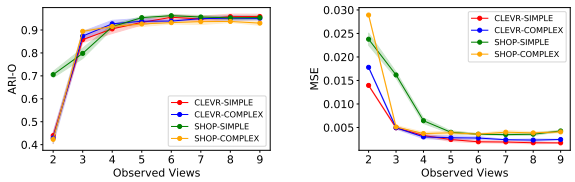


Figure 5: Curves of observed views vs different metrics in term of the single-view prediction, tested with 5 seeds. each point describes the mean value and the shadow area describes the ± standard error.

Occlusion Evaluation. Since only OLCOC in the compared methods processes the occlusion, the qualitative results of CLEVR-COMPLEX are compared in figure 3 (Left: c,d). As the camera rotates counterclockwise around the center, a gray ball is completely occluded behind the green mug in frame 2. Our model can reconstruct the complete shape of objects even when the object is completely occluded (i.e. the gray ball in frame 2). We evaluate IoU and OOA used in [Yuan et al., 2019] that respectively assess the quality of reconstructed complete shapes and the accuracy of the estimated pairwise ordering of objects. Our model clearly outperforms OCLOC, probably because OCLOC samples the pixel-wise shape during the generation, which creates noisy pixels and large shadows.

GP Prediction. Since modeling the view latent variables with GPs, we can use the analytical posterior of $\boldsymbol{z}_{\mathcal{Q}}^{\text{view}}$ to predict the remaining viewpoints given the observation set. In our setting, 10 (continuous) view representations in figure 3 (f) satisfy the GPs and then we randomly remove 4 frames (i.e. the *G.T.* of the blue is not visible). The remaining 6 frames are encoded to infer $\boldsymbol{z}_{\mathcal{T}}^{\text{obj}}$, $\boldsymbol{z}^{\text{bck}}$, $\boldsymbol{\lambda}_{\mathcal{T}}$ and $\boldsymbol{z}_{\mathcal{T}}^{\text{view}}$. 4 view representations predicted from GPs will concatenate with object-centric representations to reconstruct the 2D images. Figure 3 (f) shows that our model can predict arbitrary frames given the observation. Compared with MulMON which uses viewpoint annotations, the proposed model can additionally process the occlusion while reconstructing the novel viewpoints. To assess the segmentation performance and reconstruction quality in prediction sets, we choose 4 fixed frames with mode 1 and mode 2 to employ the prediction. Table 1 (c) and (d) show that our model is comparable to MulMON on CLEVR and clearly outperforms MulMON on SHOP. The reconstruction loss helps improve the texture characterization of objects, which is the reason why it has better MSE performances on SHOP.

Ablation Study. GPs have a general nature: as the number of observed variables increases, the prediction uncertainty gradually decreases. We consider the number of observed frames to be the most important factor (hyperparameter) that affects the accuracy and uncertainty of the prediction. To prove this, we fix a single frame and gradually increase the remaining (observed) frames from 2 to 9 (single observed view disables to infer $\boldsymbol{\lambda}_{\mathcal{Q}}$). The view representations of the predicted frame and gradually increased observed frames construct GPs together. We execute the GP prediction and plot different metrics about the predicted scene as a function of observed views. Figure 5 describes the different curves w.r.t. observed views. We find the proposed model gradually reduces the uncertainty and improves the accuracy. It tends to be stable until observed views achieve 5.

5 CONCLUSION

We propose a time-conditioned generative model for video decomposition and prediction. The model enhances the disentanglement between view and object-centric representations, and additionally uses GPs for view generation and inference. We have shown the model can: 1) aggregate 3D object-centric information and as a result, outperforms the state-of-art multi-view models. 2) restore the complete objects even when completely occluded. 3) predict unknown viewpoints without viewpoint annotations.

References

Roger E Beaty, Mathias Benedek, Paul J Silvia, and Daniel L Schacter. Creative cognition and brain network dynamics. *Trends in cognitive sciences*, 20(2):87–95, 2016.

Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in $\beta - vae$. *arXiv e-prints*, pages arXiv–1804, 2018.

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

Chang Chen, Fei Deng, and Sungjin Ahn. Roots: Object-centric representation and rendering of 3d scenes. *J. Mach. Learn. Res.*, 22:259–1, 2021.

Patrick Emami, Pan He, Sanjay Ranka, and Anand Rangarajan. Efficient iterative amortized inference for learning symmetric and disentangled multi-object representations. In *International Conference on Machine Learning*, pages 2970–2981. PMLR, 2021.

M Engelcke, AR Kosiorek, O Parker Jones, and H Posner. Genesis: Generative scene inference and sampling of object-centric latent representations. *Proceedings of the ICLR 2020*, 2020.

Martin Engelcke, Oiwi Parker Jones, and Ingmar Posner. Genesis-v2: Inferring unordered object representations without iterative refinement. *Advances in Neural Information Processing Systems*, 34:8085–8094, 2021.

SM Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Geoffrey E Hinton, et al. Attend, infer, repeat: Fast scene understanding with generative models. *Advances in Neural Information Processing Systems*, 29, 2016.

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.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015.

Jindong Jiang, Sepehr Janghorbani, Gerard De Melo, and Sungjin Ahn. Scalor: Generative world models with scalable object representations. In *International Conference on Learning Representations*, 2019.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910, 2017.

Scott P Johnson. How infants learn about the visual world. *Cognitive science*, 34(7):1158–1184, 2010.

Rishabh Kabra, Daniel Zoran, Goker Erdogan, Loic Matthey, Antonia Creswell, Matt Botvinick, Alexander Lerchner, and Chris Burgess. Simone: View-invariant, temporally-abstracted object representations via unsupervised video decomposition. *Advances in Neural Information Processing Systems*, 34:20146–20159, 2021.

Daniel Kahneman, Anne Treisman, and Brian J Gibbs. The reviewing of object files: Object-specific integration of information. *Cognitive psychology*, 24(2):175–219, 1992.

Adam Kosiorek, Hyunjik Kim, Yee Whye Teh, and Ingmar Posner. Sequential attend, infer, repeat: Generative modelling of moving objects. *Advances in Neural Information Processing Systems*, 31, 2018.

Nanbo Li, Cian Eastwood, and Robert Fisher. Learning object-centric representations of multi-object scenes from multiple views. *Advances in Neural Information Processing Systems*, 33:5656–5666, 2020.

Zhixuan Lin, Yi-Fu Wu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn. Space: Unsupervised object-oriented scene representation via spatial attention and decomposition. In *International Conference on Learning Representations*, 2019.

Zhixuan Lin, Yi-Fu Wu, Skand Peri, Bofeng Fu, Jindong Jiang, and Sungjin Ahn. Improving generative imagination in object-centric world models. In *International Conference on Machine Learning*, pages 6140–6149. PMLR, 2020.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in Neural Information Processing Systems*, 33:11525–11538, 2020.

Michal Nazarczuk and Krystian Mikolajczyk. Shop-vrb: A visual reasoning benchmark for object perception. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6898–6904. IEEE, 2020.

Daniel L Schacter, Donna Rose Addis, Demis Hassabis, Victoria C Martin, R Nathan Spreng, and Karl K Szpunar. The future of memory: remembering, imagining, and the brain. *Neuron*, 76(4):677–694, 2012.

Roger N Shepard and Jacqueline Metzler. Mental rotation of three-dimensional objects. *Science*, 171(3972):701–703, 1971.

Elizabeth S Spelke and Katherine D Kinzler. Core knowledge. *Developmental science*, 10(1):89–96, 2007.

Oliver H Turnbull, David P Carey, and ROSALEEN A McCARTHY. The neuropsychology of object constancy. *Journal of the International Neuropsychological Society*, 3(3):288–298, 1997.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial intelligence and statistics*, pages 370–378. PMLR, 2016.

Jinyang Yuan, Bin Li, and Xiangyang Xue. Generative modeling of infinite occluded objects for compositional scene representation. In *International Conference on Machine Learning*, pages 7222–7231. PMLR, 2019.

Jinyang Yuan, Bin Li, and Xiangyang Xue. Knowledge-guided object discovery with acquired deep impressions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10798–10806, 2021.

Jinyang Yuan, Bin Li, and Xiangyang Xue. Unsupervised learning of compositional scene representations from multiple unspecified viewpoints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8971–8979, 2022.