SLOTDIFFUSION: UNSUPERVISED OBJECT-CENTRIC LEARNING WITH DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Object-centric learning aims to decompose the visual data into a set of individual entities, which is distinct from traditional deep learning models that represent a scene with a global feature. Leveraging advanced architectures such as Transformer decoders, slot-based models have shown promising results in unsupervised object discovery from naturalistic inputs. In this paper, we instead focus on the slot-to-image reconstruction quality of these models, a previously overlooked topic which is important for generation tasks such as video prediction and scene editing. Despite great segmentation outputs, recent unsupervised slot models produce blurry images and temporally inconsistent videos. We address this problem by introducing slot-conditioned diffusion models due to their strong generation capacity. Our proposed method, SlotDiffusion, not only achieves better unsupervised segmentation performance, but also generates results of higher quality compared to previous state-of-the-art on both image and video datasets.

1 INTRODUCTION

Humans perceive the world with discrete concepts such as objects and events (Spelke & Kinzler, 2007), which can be processed independently and composed to support systematic generalization of intelligence (Greff et al., 2020). Similarly, object-centric learning that aims to equip machines with such structured representation also has the potential to improve the generalizability, robustness and interpretability of AI algorithms (Lake et al., 2017; Schölkopf et al., 2021). For example, explicit decomposition of scenes into objects facilitates visual reasoning tasks (Chen et al., 2020; Ding et al., 2021b;a). Also, capturing the compositional structure of world is useful for image generation (Singh et al., 2021; Sylvain et al., 2021) and future prediction (Ye et al., 2019; Wu et al., 2022).

Due to its practical implications, unsupervised object discovery from visual data has been a long-standing problem in computer vision. Earlier attempts focus on synthetic data (Johnson et al., 2017; Yi et al., 2019) and bake in strong priors in their frameworks (Jiang et al., 2019; Lin et al., 2020), preventing them from scaling to more complex scenes. Later works generalize the Scaled Dot-Product Attention (Vaswani et al., 2017) and propose the Slot Attention mechanism (Locatello et al., 2020), which eliminates domain-specific priors (Kipf et al., 2021; Singh et al., 2022). To work on real-world data, recent approaches introduce additional supervision signals such as optical flow (Yang et al., 2021), depth (Elsayed et al., 2022) and pre-trained feature encoder (Seitzer et al., 2022).

Despite tremendous progress in object segmentation, we argue that the generation capacity of slot-based models is underexplored. Wu et al. (2022) shows that, while the autoregressive (AR) Transformer-based decoder enables STEVE (Singh et al., 2022) to handle more complex videos, its slot-to-image reconstruction quality is worse than the naive CNN-based decoder in SAVi (Kipf et al., 2021), which hinders its application to generation tasks. In this paper, we propose SlotDiffusion, an unsupervised object-centric model with a slot-conditioned diffusion model (DM) (Ho et al., 2020) decoder. Thanks to the strong capacity of DMs, SlotDiffusion achieves a better trade-off between segmentation and reconstruction compared to previous state-of-the-art on 4 datasets.

2 BACKGROUND: UNSUPERVISED OBJECT-CENTRIC LEARNING

The goal of unsupervised object-centric learning is to represent the scene with a set of object *slots* without instance-level supervision. Here, we review the SAVi family which builds on the Slot Attention (Locatello et al., 2020) operation and runs on videos in a recurrent encoder-decoder manner.

Given T input frames $\{\boldsymbol{x}_t\}_{t=1}^T$, SAVi first leverages a per-frame image encoder to extract features, adds positional encodings, and flattens them into a set of vectors $\boldsymbol{h}_t = f_{\text{enc}}(\boldsymbol{x}_t) \in \mathbb{R}^{M \times D_{\text{enc}}}$. Then, the model initializes N slots $\tilde{\mathcal{S}}_t \in \mathbb{R}^{N \times D_{\text{slot}}}$ from a set of learnable vectors ($t = 1$), and updates them with Slot Attention as $\mathcal{S}_t = f_{\text{SA}}(\tilde{\mathcal{S}}_t, \boldsymbol{h}_t)$. f_{SA} performs soft feature clustering, where slots compete with each other to capture certain area of the input via iterative cross-attention (Vaswani et al., 2017). To achieve temporally aligned slots, SAVi leverages a Transformer-based predictor to initialize $\tilde{\mathcal{S}}_t$ ($t \geq 2$) as $\tilde{\mathcal{S}}_t = f_{\text{pred}}(\mathcal{S}_{t-1})$. Finally, these models use a decoder f_{dec} to reconstruct the input $\boldsymbol{x}_t$ from slots $\mathcal{S}_t$ as training signal. See Figure 4 in the Appendix for the model pipeline.

Mixture-based decoder. In vanilla SAVi (Kipf et al., 2021), f_{dec} consists of a stack of up-sampling deconvolution layers. It decodes each slot $\mathcal{S}_t^i$ to an RGB image $\boldsymbol{y}_t^i$ and an alpha mask $\boldsymbol{m}_t^i$, which are combined into the final reconstructed image $\hat{\boldsymbol{x}}_t$. The training loss is simply a reconstruction MSE:

$$(\boldsymbol{y}_t^i, \boldsymbol{m}_t^i) = f_{\text{dec}}^{\text{mix}}(\mathcal{S}_t^i), \;\; \hat{\boldsymbol{x}}_t = \sum_{i=1}^{N} \boldsymbol{m}_t^i \odot \boldsymbol{y}_t^i, \;\; \mathcal{L}_{\text{image}} = \sum_{t=1}^{T} ||\boldsymbol{x}_t - \hat{\boldsymbol{x}}_t||^2. \tag{1}$$

Transformer-based decoder. The above mixture-based decoder has limited modeling capacity as it decodes each slot separately without interactions. Also, pixel-level reconstruction biases the model to low-level color statistics, which only proves effective on objects with uniform colors, and cannot scale to complex data with textured objects. Current state-of-the-art model STEVE (Singh et al., 2022) thus proposes to reconstruct intermediate features produced by a trained network (Singh et al., 2021). Given frame $\boldsymbol{x}_t$, STEVE leverages a dVAE encoder to convert it into a sequence of patch tokens $\boldsymbol{o}_t = \{\boldsymbol{o}_t^i\}_{i=1}^L$, which serve as the reconstruction targets for the AR Transformer decoder:

$$\boldsymbol{o}_t = f_{\text{enc}}^{\text{dVAE}}(\boldsymbol{x}_t), \;\; \hat{\boldsymbol{o}}_t^l = f_{\text{dec}}^{\text{trans}}(\mathcal{S}_t; \boldsymbol{o}_t^1, ..., \boldsymbol{o}_t^{l-1}), \;\; \mathcal{L}_{\text{token}} = \sum_{t=1}^{T} \sum_{l=1}^{L} \text{CrossEntropy}(\boldsymbol{o}_t^l, \hat{\boldsymbol{o}}_t^l). \tag{2}$$

Thanks to the cross-attention mechanism in Transformer decoder and the feature-level reconstruction objective, STEVE succeeds on naturalistic videos with textured objects and background.

3 METHOD

Object-centric generative models (Singh et al., 2021; Zoran et al., 2021) often decompose the generation process to first predicting the object slots, followed by decoding slots back to the pixel space. As shown in Wu et al. (2022), the generation quality is largely bounded by the slot decoder. STEVE's Transformer-based decoder produces low-quality results due to two reasons: i) treating images as sequences of tokens ignores their spatial structure; ii) autoregressive token prediction causes severe error accumulation. We overcome these drawbacks by introducing diffusion models as the slot decoder, which preserve the spatial dimension of images, and iteratively refine the generation results.

Diffusion model. DMs (Sohl-Dickstein et al., 2015; Ho et al., 2020) are probabilistic models that learn a data distribution $p_\theta(\boldsymbol{X}_0)$ by gradually denoising a standard Gaussian distribution, in the form of $p_\theta(\boldsymbol{X}_0) = \int p_\theta(\boldsymbol{X}_{0:T})\, d\boldsymbol{X}_{1:T}$, where $\boldsymbol{X}_{1:T}$ are intermediate denoising results. The forward process of DMs is a Markov Chain that adds Gaussian noise to the clean data $\boldsymbol{X}_0$, which is controlled by a pre-defined variance schedule $\{\beta_t\}_{t=1}^T$. Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, we have:

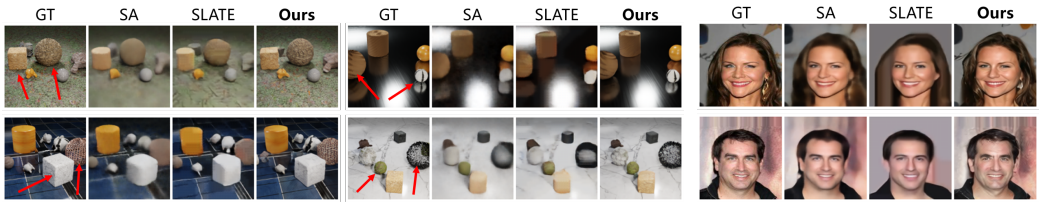
$$q(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}) = \mathcal{N}(\boldsymbol{X}_t|\sqrt{1-\beta_t}\boldsymbol{X}_{t-1}, \beta_t \boldsymbol{I}) \;\Rightarrow\; q(\boldsymbol{X}_t|\boldsymbol{X}_0) = \mathcal{N}(\boldsymbol{X}_t|\sqrt{\bar{\alpha}_t}\boldsymbol{X}_0, (1-\bar{\alpha}_t)\boldsymbol{I}). \tag{3}$$

During training, a network $\epsilon_\theta(\boldsymbol{X}_t, t)$ is trained to predict the noise applied to a noisy sample:

$$\boldsymbol{X}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{X}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t, \;\; \mathcal{L}_{\text{DM}} = ||\epsilon_t - \epsilon_\theta(\boldsymbol{X}_t, t)||^2, \text{ where } \epsilon_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}). \tag{4}$$

At inference time, we can start from a random Gaussian noise, and apply the trained denoiser to iteratively refine the sample. See Appendix C for detailed formulation of diffusion models.

SlotDiffusion. Our model consists of the same image encoder, Slot Attention module and slot predictor as SAVi and STEVE, while only replacing the slot decoder with the DM-based one. Inspired by STEVE and Latent Diffusion Model (LDM) (Rombach et al., 2022), we train our DM decoder to denoise features in the latent space. This improves the segmentation results with higher-level reconstruction target, and greatly reduce the training cost. Specifically, we pre-train a VQ-VAE (Razavi et al., 2019) to extract feature maps $\boldsymbol{z} \in \mathbb{R}^{h \times w \times D_{\text{vq}}}$ from $\boldsymbol{x}$ before training SlotDiffusion. From now on we omit t as video timestamps, and will use it to only represent diffusion steps in DM.



(a) CLEVRTex textured objects and scenes (b) CelebA human face

Figure 1: Image reconstruction results on both datasets. Note the object textures and human hairs.

To condition the decoder on slots $\mathcal{S}$, we notice that slots are N 1D feature vectors, which are similar to text embeddings output by language models. Therefore, we follow text-guided LDM to guide the denoising process via cross-attention as $\boldsymbol{o} = \text{CrossAttention}(Q(\tilde{\boldsymbol{o}}), K(\mathcal{S}), V(\mathcal{S}))$. Here, Q, K, V are learnable linear projections, $\tilde{\boldsymbol{o}}$ is an intermediate feature map from ϵ_θ, and $\boldsymbol{o}$ is the feature map fused with slots information. In practice, we perform conditioning after several layers in the DM decoder. Overall, our model is trained with a slot-conditioned denoising loss over VQ-VAE features:

$$\boldsymbol{z}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{z} + (1-\bar{\alpha}_t)\epsilon_t, \;\; \mathcal{L}_{\text{slot}} = ||\epsilon_t - \epsilon_\theta(\boldsymbol{z}_t, t, \mathcal{S})||^2, \text{ where } \epsilon_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}). \tag{5}$$

4 Experiments

SlotDiffusion is general unsupervised object-centric learning framework that can be applied to both image and video datasets. We evaluate our model on two image and two video datasets in terms of reconstruction error (generation power) and segmentation results (scene decomposition quality).

4.1 Experimental Setup

We briefly introduce the experimental setup here, which is detailed in Appendix D.

Datasets. We select the two most complex image datasets from SLATE (Singh et al., 2021), namely, CLEVRTex (Karazija et al., 2021) and CelebA (Liu et al., 2015). For video datasets, we follow STEVE (Singh et al., 2022) and SAVi (Kipf et al., 2021) to use MOVi-D/E (Greff et al., 2022).

Metrics. To evaluate the generation quality, we adopt the mean squared error (MSE) between the visual inputs and the reconstructions decoded from slots. We also compute the VGG perceptual distance (LPIPS) (Zhang et al., 2018). For segmentation results, we measure the FG-ARI and mIoU which are two widely used metrics in unsupervised object-centric learning papers.

Baselines. We compare SlotDiffusion with state-of-the-art fully unsupervised object-centric models. On image datasets, we adopt Slot Attention (SA) (Locatello et al., 2020) and SLATE (Singh et al., 2021). On video datasets we adopt SAVi (Kipf et al., 2021) and STEVE (Singh et al., 2022). They are representative models which use the mixture-based decoder and the Transformer-based decoder.

Our implementation details. We use the same image encoder, Slot Attention module, and transition predictor as baselines, while only replacing the slot decoder with the conditional LDM. We first pre-train VQ-VAE on each dataset, and then freeze it and train the object-centric model with Eq. (5).

4.2 Results on Image Datasets

Method	CLEVRTex				CelebA	
	MSE ↓	LPIPS ↓	FG-ARI ↑	mIoU ↑	MSE ↓	LPIPS ↓
SA	**212.4**	0.410	67.92	36.70	**243.3**	0.284
SLATE	313.3	0.391	67.50	38.14	744.8	0.324
Ours	237.5	**0.126**	68.39	**41.66**	439.0	**0.212**

Table 1: Evaluation results on image datasets. SA stands for Slot Attention. FG-ARI and mIoU numbers are in %.

Table 1 presents the results. For reconstruction quality, SlotDiffusion outperforms both baselines with a sizeable margin in LPIPS, and achieves the second lowest MSE. As discussed in Zhang et al. (2018), LPIPS aligns well with human perception, while MSE is a poor metric as it favors blurry results. This can be verified by the qualitative results in Figure 1, where SA and SLATE both reconstruct over-smoothing images with distorted object attributes. On the contrary, our DM decoder is able to iteratively refine the results, leading to accurate textures and details such as hairs.

For scene decomposition, we only evaluate on CLEVRTex as CelebA does not provide object mask annotations. All three methods achieve similar FG-ARI score, but we show clear advancement in terms of mIoU. This is because SlotDiffusion segments objects from background sharply. In contrast, the baselines introduce lots of false positives by assigning background pixels to objects.

Method	MOVi-D				MOVi-E			
	MSE ↓	LPIPS ↓	FG-ARI ↑	mIoU ↑	MSE ↓	LPIPS ↓	FG-ARI ↑	mIoU ↑
SAVi	**430.1**	0.519	34.59	10.57	**447.1**	0.533	41.36	13.53
STEVE	655.4	0.549	48.18	20.41	588.1	0.494	53.69	18.69
Ours	616.8	**0.384**	**51.80**	**21.55**	577.4	**0.370**	**59.63**	**20.38**

Table 2: Evaluation results on MOVi-D and MOVi-E video datasets. FG-ARI and mIoU numbers are in %.

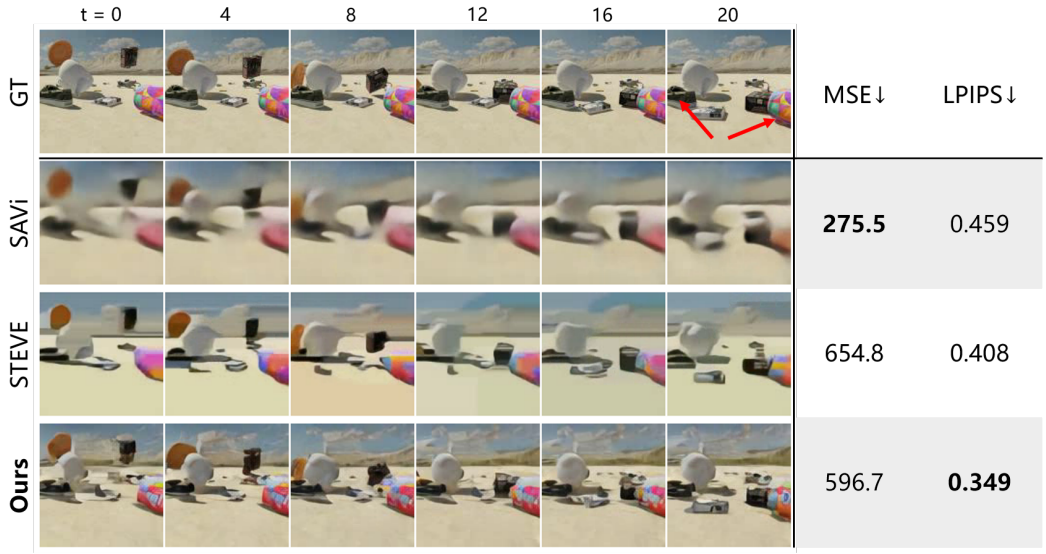


Figure 2: Qualitative results of slot-to-video reconstruction on MOVi-E. On the right, we report the metrics of the visualized videos for each model. Despite a low MSE, the SAVi results are very blurry.

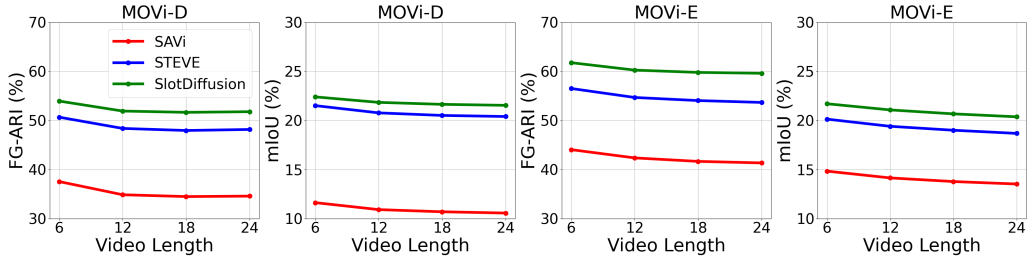


Figure 3: Segmentation results as a function of video length. We show FG-ARI and mIoU on both datasets.

4.3 Results on Video Datasets

We show the quantitative results on both video datasets in Table 2. For reconstruction quality, SlotDiffusion still achieves state-of-the-art LPIPS. Figure 2 presents the qualitative results, where SAVi and SLATE produce videos with blurry objects and backgrounds. Thanks to the powerful DM, our model retains the detailed shapes and textures of the objects. Also, our generated object properties are temporally more consistent. See Appendix E.2 for more qualitative results.

For video segmentation, SlotDiffusion achieves both the best FG-ARI and mIoU. Figure 3 shows how the segmentation results change with video length, which measures the object tracking performance. Despite trained only on video clips of length 3, our model is able to generalize to the entire videos at test time, outperforming all the baselines consistently. See Appendix E.2 for visualizations.

5 Conclusion

In this paper, we propose SlotDiffusion by incorporating diffusion models with object-centric models. Conditioned on object slots, our LDM-based decoder performs iterative denoising over latent image features, providing strong learning signal for unsupervised scene decomposition. Experimental results on both image and video datasets demonstrate that our model can achieve state-of-the-art segmentation results and slot-to-image reconstruction quality. We discuss the limitations and potential future directions of this work in Appendix F.

REFERENCES

Daniel Bear, Elias Wang, Damian Mrowca, Felix Jedidja Binder, Hsiao-Yu Tung, RT Pramod, Cameron Holdaway, Sirui Tao, Kevin A Smith, Fan-Yun Sun, et al. Physion: Evaluating physical prediction from vision in humans and machines. In *NeurIPS Datasets and Benchmarks Track*, 2021.

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.

Zhenfang Chen, Jiayuan Mao, Jiajun Wu, Kwan-Yee Kenneth Wong, Joshua B Tenenbaum, and Chuang Gan. Grounding physical concepts of objects and events through dynamic visual reasoning. In *ICLR*, 2020.

Antonia Creswell, Rishabh Kabra, Chris Burgess, and Murray Shanahan. Unsupervised object-based transition models for 3d partially observable environments. *NeurIPS*, 34, 2021.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *NeurIPS*, 34:8780–8794, 2021.

David Ding, Felix Hill, Adam Santoro, Malcolm Reynolds, and Matt Botvinick. Attention over learned object embeddings enables complex visual reasoning. *NeurIPS*, 34, 2021a.

Mingyu Ding, Zhenfang Chen, Tao Du, Ping Luo, Josh Tenenbaum, and Chuang Gan. Dynamic visual reasoning by learning differentiable physics models from video and language. *NeurIPS*, 34, 2021b.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. *arXiv preprint arXiv:2204.11918*, 2022.

Gamaleldin Fathy Elsayed, Aravindh Mahendran, Sjoerd van Steenkiste, Klaus Greff, Michael Curtis Mozer, and Thomas Kipf. Savi++: Towards end-to-end object-centric learning from real-world videos. *NeurIPS*, 2022.

Martin Engelcke, Adam R Kosiorek, Oiwi Parker Jones, and Ingmar Posner. Genesis: Generative scene inference and sampling with object-centric latent representations. In *ICLR*, 2019.

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

Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. In *ICML*, pp. 2424–2433. PMLR, 2019.

Klaus Greff, Sjoerd Van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *CVPR*, pp. 3749–3761, 2022.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pp. 16000–16009, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33: 6840–6851, 2020.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *NeurIPS*, 2022b.

Tianyu Hua, Yonglong Tian, Sucheng Ren, Hang Zhao, and Leonid Sigal. Self-supervision through random segments with autoregressive coding (randsac). *arXiv preprint arXiv:2203.12054*, 2022.

Jindong Jiang, Sepehr Janghorbani, Gerard De Melo, and Sungjin Ahn. Scalor: Generative world models with scalable object representations. In *ICLR*, 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 *CVPR*, pp. 2901–2910, 2017.

Laurynas Karazija, Iro Laina, and Christian Rupprecht. Clevrtex: A texture-rich benchmark for unsupervised multi-object segmentation. In *NeurIPS Datasets and Benchmarks Track*, 2021.

Thomas Kipf, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. In *ICLR*, 2021.

Adam Kosiorek, Hyunjik Kim, Yee Whye Teh, and Ingmar Posner. Sequential attend, infer, repeat: Generative modelling of moving objects. *NeurIPS*, 31, 2018.

Jannik Kossen, Karl Stelzner, Marcel Hussing, Claas Voelcker, and Kristian Kersting. Structured object-aware physics prediction for video modeling and planning. In *ICLR*, 2019.

Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.

Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. *arXiv preprint arXiv:2211.10440*, 2022.

Zhixuan Lin, Yi-Fu Wu, Skand Peri, Bofeng Fu, Jindong Jiang, and Sungjin Ahn. Improving generative imagination in object-centric world models. In *ICML*, pp. 6140–6149. PMLR, 2020.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

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

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. In *NeurIPS*, 2022.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, pp. 8162–8171. PMLR, 2021.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *NeurIPS*, 32, 2019.

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

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pp. 234–241. Springer, 2015.

Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pp. 1–10, 2022a.

Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *TPAMI*, 2022b.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, et al. Bridging the gap to real-world object-centric learning. *arXiv preprint arXiv:2209.14860*, 2022.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

Gautam Singh, Fei Deng, and Sungjin Ahn. Illiterate dall-e learns to compose. In *ICLR*, 2021.

Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple unsupervised object-centric learning for complex and naturalistic videos. *arXiv preprint arXiv:2205.14065*, 2022.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pp. 2256–2265. PMLR, 2015.

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

Tristan Sylvain, Pengchuan Zhang, Yoshua Bengio, R Devon Hjelm, and Shikhar Sharma. Object-centric image generation from layouts. In *AAAI*, volume 35, pp. 2647–2655, 2021.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017.

Nicholas Watters, Loic Matthey, Christopher P Burgess, and Alexander Lerchner. Spatial broadcast decoder: A simple architecture for learning disentangled representations in vaes. *arXiv preprint arXiv:1901.07017*, 2019.

Xin Wen, Bingchen Zhao, Anlin Zheng, Xiangyu Zhang, and Xiaojuan Qi. Self-supervised visual representation learning with semantic grouping. *NeurIPS*, 2022.

Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models. *arXiv preprint arXiv:2210.05861*, 2022.

Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised video object segmentation by motion grouping. In *ICCV*, pp. 7177–7188, 2021.

Yufei Ye, Maneesh Singh, Abhinav Gupta, and Shubham Tulsiani. Compositional video prediction. In *ICCV*, pp. 10353–10362, 2019.

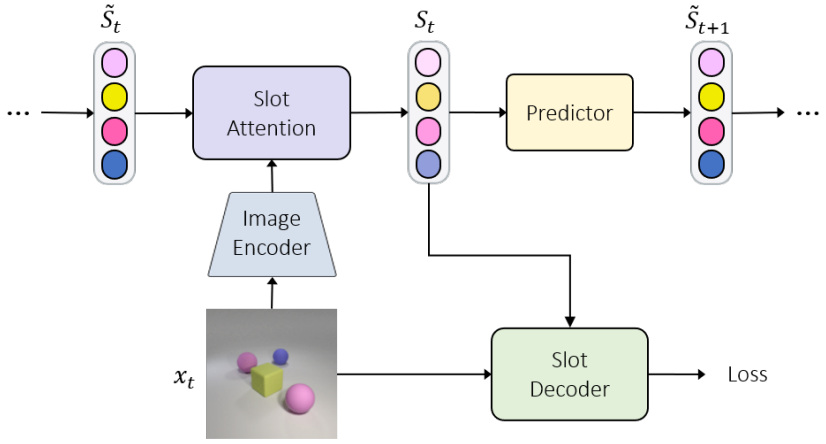
Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. In *ICLR*, 2019.

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

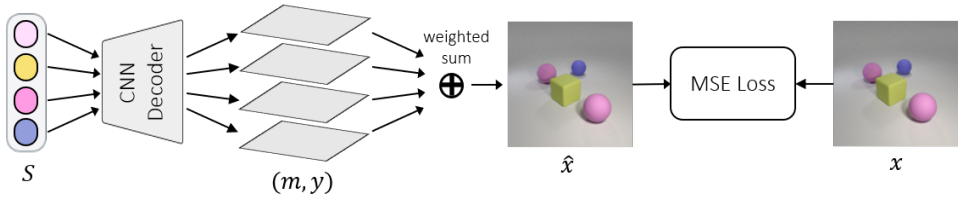
Daniel Zoran, Rishabh Kabra, Alexander Lerchner, and Danilo J Rezende. Parts: Unsupervised segmentation with slots, attention and independence maximization. In *ICCV*, pp. 10439–10447, 2021.

A Model Architecture

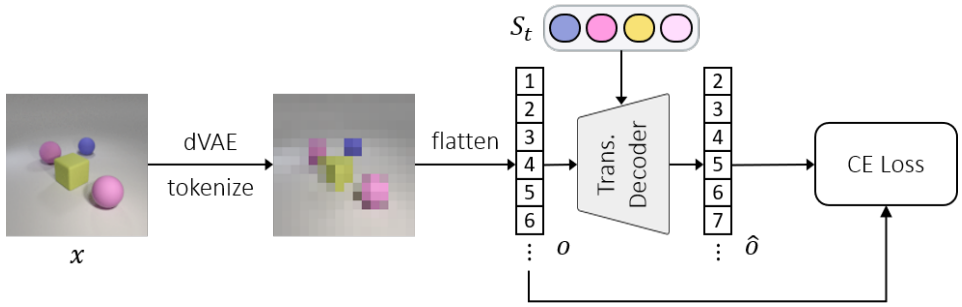
Figure 4 shows the general pipeline of video Slot Attention models, and compare two existing slot decoder with our proposed decoder.



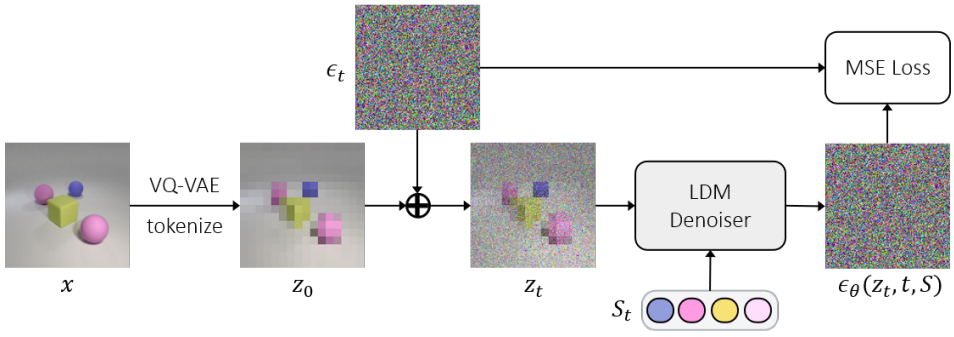
(a) General pipeline of video-based Slot Attention models.



(b) Mixture-based CNN decoder.



(c) Transformer-based autoregressive decoder.



(d) Latent diffusion model based decoder.

Figure 4: Illustration of (a) the training pipeline of video Slot Attention model, (b) the mixture-based decoder used in SAVi (Kipf et al., 2021), (c) the Transformer-based decoder used in STEVE (Singh et al., 2022), and (d) the LDM-based decoder proposed by us.

B Additional Related Work

Unsupervised object-centric learning from images. Our work is directly related to research that aim at learning to represent the visual data with a set of feature vectors without explicit supervision. Earlier attempts starts from synthetic image datasets with well-defined objects (Eslami et al., 2016; Burgess et al., 2019; Greff et al., 2019; Engelcke et al., 2019; Locatello et al., 2020). They typically perform iterative inference to extract object-centric features from images, followed by a mixture-based CNN decoder applied to each slot separately for reconstruction. For example, AIR (Eslami et al., 2016) uses a patch-based decoder to decode each object locally, and transform them back to form the original image. MONet (Burgess et al., 2019) and Slot Attention (Locatello et al., 2020) both adopt the spatial broadcast decoder (Watters et al., 2019) to predict an RGB image and an objectness mask from each slot, and combine them via alpha masking. Recently, SLATE (Singh et al., 2021) challenges the traditional design with a Transformer-based decoder, which helps the model scale to more complex data. Another line of works (Wen et al., 2022; Seitzer et al., 2022) focus on contrastive representation learning for object discovery without decoding. However, they are not applicable for generation tasks as they cannot generate images from slots.

Unsupervised object-centric learning from videos. Compared to images, videos provide additional information such as motion cues. Our work also builds upon recent efforts in decomposing raw videos into temporally aligned object slots (Kosiorek et al., 2018; van Steenkiste et al., 2018; Kossen et al., 2019; Lin et al., 2020; Kipf et al., 2021; Zoran et al., 2021). These works usually integrate a dynamics module to model the interactions between objects compared to their image-based counterparts. For example, STOVE (Kossen et al., 2019) adopts a Graph Neural Network (GNN), while OAT (Creswell et al., 2021), SAVi (Kipf et al., 2021) and PARTS (Zoran et al., 2021) leverage the powerful Transformer architecture. However, the limited modeling capacity of their decoders still prevents them from scaling to more realistic videos. Some works (Elsayed et al., 2022) thus introduce additional supervision such as optical flow and depth. Recently, STEVE (Singh et al., 2022) pushes the limit of fully unsupervised object-centric learning by using the Transformer decoder from SLATE. Nevertheless, the autoregressive generation mechanism of Transformer degrades its generation quality, which we aim to solve in this paper.

Diffusion model. Recently, diffusion models have achieved tremendous progress in generation tasks, including images (Ho et al., 2020; Nichol & Dhariwal, 2021; Dhariwal & Nichol, 2021), videos (Ho et al., 2022b; Singer et al., 2022; Ho et al., 2022a), and 3D shapes (Poole et al., 2022; Lin et al., 2022), showing their great ability in sample quality and conditioning. The generative process of diffusion models is formulated as an iterative denoising procedure with a denoising network, usually implemented as a U-Net (Ronneberger et al., 2015). However, the memory consumption of DM scales quadratically with the input resolution due to the use of self-attention layers in the U-Net. To reduce the training cost, LDM (Rombach et al., 2022) proposes to run the diffusion process in the latent space of a pre-trained auto-encoder. LDM also introduces a flexible conditioning mechanism via cross-attention between U-Net feature maps and conditional inputs. In this work, we adopt the slot-conditioned LDM as our decoder for both better segmentation and reconstruction quality.

C Details on Diffusion Models

Diffusion models are probabilistic models that learn a data distribution $p_\theta(\boldsymbol{X}_0)$ by gradually denoising a standard Gaussian distribution, in the form of $p_\theta(\boldsymbol{X}_0) = \int p_\theta(\boldsymbol{X}_{0:T})\, d\boldsymbol{X}_{1:T}$. Here, $\boldsymbol{X}_{1:T}$ are intermediate denoising results with the same shape as the clean data $\boldsymbol{X}_0 \sim q(\boldsymbol{X})$, and θ are learnable parameters of the denoising U-Net model.

The joint distribution $q(\boldsymbol{X}_{1:T}|\boldsymbol{X}_0)$ is called the *forward process* or *diffusion process*, which is a fixed Markov Chain that gradually adds Gaussian noise to $\boldsymbol{X}_0$. The noise is controlled by a pre-defined variance schedule $\{\beta_t\}_{t=1}^T$:

$$q(\boldsymbol{X}_{1:T}|\boldsymbol{X}_0) = \prod_{t=1}^{T} q(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}),\;\; q(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}) = \mathcal{N}(\boldsymbol{X}_t|\sqrt{1-\beta_t}\boldsymbol{X}_{t-1}, \beta_t\boldsymbol{I}). \tag{6}$$

Thanks to the nice property of Gaussian distributions, $\boldsymbol{X}_t$ can be sampled directly from $\boldsymbol{X}_0$ in closed form without adding the noise t times. Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, we have:

$$q(\boldsymbol{X}_t|\boldsymbol{X}_0) = \mathcal{N}(\boldsymbol{X}_t|\sqrt{\bar{\alpha}_t}\boldsymbol{X}_0, (1-\bar{\alpha}_t)\boldsymbol{I}) \;\Rightarrow\; \boldsymbol{X}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{X}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t, \text{ where } \epsilon_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}). \tag{7}$$

We can now train a model to reverse this process and thus generate target data from random noise $\boldsymbol{X}_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. The *reverse process* $p_\theta(\boldsymbol{X}_{0:T})$ is also defined as a Markov Chain with a learned Gaussian transition:

$$p_\theta(\boldsymbol{X}_{0:T}) = p(\boldsymbol{X}_T)\prod_{t=1}^{T} p_\theta(\boldsymbol{X}_{t-1}|\boldsymbol{X}_t), \;\; p_\theta(\boldsymbol{X}_{t-1}|\boldsymbol{X}_t) = \mathcal{N}(\boldsymbol{X}_{t-1}|\mu_\theta(\boldsymbol{X}_t, t), \Sigma_\theta(\boldsymbol{X}_t, t)) \quad (8)$$

In practice, we do not learn the variance and usually set it to $\Sigma_t = \beta_t \boldsymbol{I}$ or $\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t \boldsymbol{I}$ since it leads to unstable training (Ho et al., 2020). Also, instead of learning the mean μ_θ directly, we learn to predict the noise ϵ_t in Equation (6). See Ho et al. (2020) for how we can sample $\boldsymbol{X}_{t-1}$ given $\boldsymbol{X}_t$ and the predicted ϵ_t at the inference stage.

The training process of diffusion models is thus straightforward given Equation (7). At each training step, we sample a batch of clean data $\boldsymbol{X}_0$ from the training set, timesteps t uniformly from $\{1, ..., T\}$, and random Gaussian noise $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. We then create the noisy version of data $\boldsymbol{X}_t$ by applying Equation (7). A denoising model ϵ_θ is trained to predict the noise with an MSE loss:

$$\mathcal{L}_{DM} = \mathbb{E}_{\boldsymbol{X}, t, \epsilon_t}[||\epsilon_t - \epsilon_\theta(\boldsymbol{X}_t, t)||^2] \quad (9)$$

D Detailed Experimental Setup

In this section, we detail the datasets, evaluation metrics, baseline methods used in the experiments, and the implementation of our model.

D.1 Datasets

CLEVRTex (Karazija et al., 2021). This dataset augments the CLEVR (Johnson et al., 2017) dataset with more diverse object shapes, materials and textures. The backgrounds in CLEVRTex also present complex textures compared to the plain gray one in CLEVR. Therefore, this dataset is visually much more complex than CLEVR. As shown in their paper, only 3 out of 10 benchmarked unsupervised object-centric models can achieve an mIoU higher than 30%. We train our model on the training set consisting of 40k images, and test on the test set with 5k samples. We use the same data pre-processing steps, i.e. first center-crop to 192×192, and then resize to 128×128.

CelebA (Liu et al., 2015). This dataset contains over 200k real-world celebrity images. All images are mostly occupied by human faces, covering a large variation of poses and backgrounds. This dataset is more challenging compared to previous synthetic datasets as real-world images typically have unconstrained background clutters and complicated lighting conditions. We train our model on the training set with around 160k images, and test on the test split with around 20k images. For data pre-processing, we simply resize all images to 128×128.

MOVi-D/E (Greff et al., 2022). MOVi-D and MOVi-E are the 2 most challenging versions from the MOVi benchmark generated using the Kubric simulator. Their videos feature photo-realistic backgrounds and real-world objects from the Google Scanned Objects (GSO) repository (Downs et al., 2022), where one or several objects are thrown to the ground to collide with other objects. Compared to MOVi-D, MOVi-E applies linear camera motion. We follow the official train-test split to evaluate our model. For data pre-processing, we simply resize all frames to 128×128.

D.2 Evaluation Metrics

To evaluate the generation quality of SlotDiffusion, we compute the mean squared error (MSE) of the images reconstructed from the object slots. Following prior works, we scale the images to $[0, 1]$, and sum the errors over channel and spatial dimensions. As pointed out by Zhang et al. (2018), MSE does not align well with human perception about visual quality as it favors over-smooth results. Therefore, we additionally compute the perceptual distance (LPIPS) (Zhang et al., 2018) metric.

To evaluate the scene decomposition results, we compute the FG-ARI and mIoU of the object masks. FG-ARI is widely used in previous object-centric learning papers, which only consider the foreground objects. As suggested by Engelcke et al. (2019); Karazija et al. (2021), we should also compute the mIoU which evaluates background segmentation. It is worth noting that on video datasets, we flatten the temporal dimension into spatial dimensions when computing the metrics, thus taking the temporal consistency of object masks into consideration. Being able to consistently track all the objects is also an important property of video slot models.

Dataset	CLEVRTex	CelebA	MOVi-D/E
Number of Slots N	11	4	15
Slot Size D_{slot}	192	192	192
Slot Attention Iteration	3	3	2
Max Learning Rate	2e-4	2e-4	1e-4
Gradient Clipping	1	None	0.05
Batch Size	64	64	32
Training Epochs	100	50	30

Table 3: Variations in model architectures and training settings on different datasets.

		SAVi	STEVE	SlotDiffusion
Train	Memory (GB)	32	51	24
	Time (s)	0.57	0.87	0.77
Test	Time (min)	0.7	226	7

Table 4: Comparison of model complexity on the MOVi-D/E video datasets. We measure the training memory consumption, time per training step and generation time of 100 videos at test stage. For training, we report the default settings (batch size 32 of length-3 video clips, frame resolution 128×128) on NVIDIA A40 GPUs. For testing, we report the inference time on NVIDIA T4 GPUs.

D.3 Baselines

We adopt Slot Attention and SAVi as representative models which use a mixture-based CNN decoder, and SLATE and STEVE for models with a autoregressive Transformer-based decoder. Since we augment SlotDiffusion with a stronger ResNet18 encoder compared to the previous stacked CNN encoder, we also re-train baselines with the same encoder, which achieves better performance than reported in their papers. We found that using a larger decoder and training longer leads to better performance, and thus report the baseline results with the best hyper-parameters we discovered.

D.4 Implementation Details of SlotDiffusion

We use the same VQ-VAE (Razavi et al., 2019) architecture for all datasets, which is adopted from LDM (Rombach et al., 2022). We use 3 encoder and decoder blocks, resulting in 4x down-sampling of the feature maps $\boldsymbol{z}$ compared to input images $\boldsymbol{x}$. We pre-train the VQ-VAE for 100 epochs on each datasets with a cosine learning rate schedule and fix it during the object-centric model training.

For the object-centric model, we only replace the decoder with LDM (Rombach et al., 2022) compared to Slot Attention on image datasets and SAVi on video datasets. Another difference we made is to use a modified ResNet18 encoder (Kipf et al., 2021) to extract image features. For the LDM-based slot decoder, the training target is to predict the noise ϵ added to the features produced by a pre-trained VQ-VAE. Following prior works, the denoising network $\epsilon_\theta(\boldsymbol{z}_t, t, \mathcal{S})$ is implemented as a U-Net (Ronneberger et al., 2015) with global self-attention operation in each block. We use the same noise schedule $\{\beta_t\}_{t=1}^T$ and U-Net hyper-parameters as Rombach et al. (2022). See Table 3 for detailed slot configurations and training settings.

Diffusion model is notoriously slow in doing generation due to the iterative denoising process, where the diffusion step T is often 1,000. Fortunately, researchers have developed several methods to accelerate this procedure. We employ the DPM-Solver (Lu et al., 2022) which reduce the sampling steps to 20. Therefore, our reconstruction speed is even faster than models with Transformer decoder, as they need to forward their model number of tokens (typically 1,024) times, while we only need 20 times, and can get better reconstruction quality.

E Additional Experimental Results

E.1 Computational Requirements

We empirically show the speed and GPU memory requirement at training time, as well as the time required to reconstruct 100 videos at test time of SAVi, STEVE and SlotDiffusion in Table 4. Interestingly, our model requires the least GPU memory during training despite achieving the best performance. This is because we run the diffusion process at latent space, whose spatial dimension is only $1/4$ of the input resolution. On the other hand, SAVi applies CNN to directly reconstruct

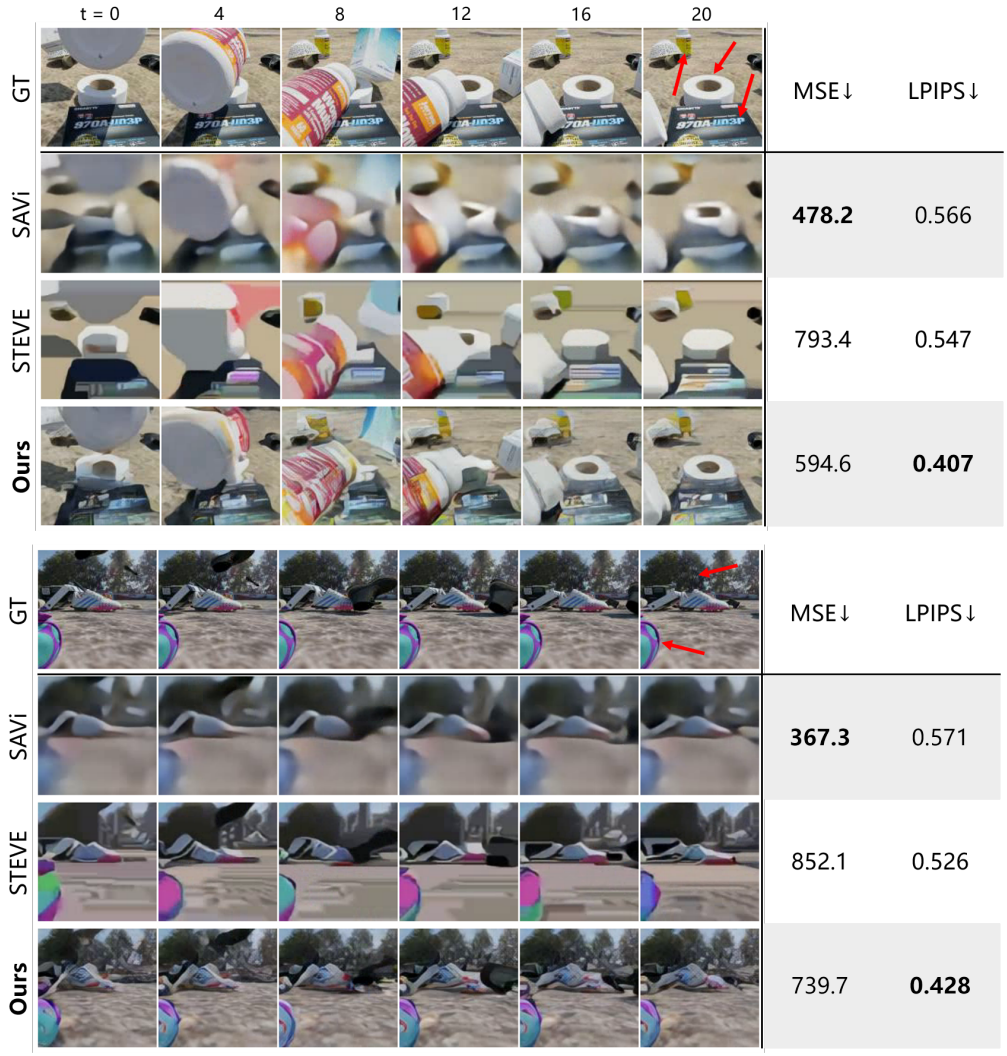


Figure 5: Qualitative results of video reconstructions on MOVi-D datasets.

the images at original resolution, and STEVE uses Transformer to predict a long sequence of 1,024 tokens. Both of these designs consumes large GPU memory. In terms of training and generation speed, SlotDiffusion ranks second. SAVi runs extremely fast since it decodes images in one-shot, while STEVE and our model both need to do iterative sampling. Thanks to the efficient DPM-Solver sampler, we only require 20 times forward pass, while STEVE requires 1,024 times.

E.2 Additional Qualitative Results

Video reconstruction. Figure 5 and Figure 6 show video reconstruction results on MOVi-D and MOVi-E datasets, respectively. Compared to baselines that produce blurry frames, SlotDiffusion is able to preserve the local appearances such as textures on the objects and backgrounds. However, there is still large room for improvement in finer details. For example, in Figure 5 (top), we cannot reconstruct the texts on the object surfaces. Also, in Figure 6 (bottom), we fail to retain the smooth round opening of the red object.

Video segmentation. We show the video segmentation results on both video datasets in Figure 7 and Figure 8. As observed in previous work (Singh et al., 2022), MOVi-D is more challenging than MOVi-E since most of the objects are static, while on MOVi-E the moving camera provides motion cues. Indeed, the static objects on MOVi-D are usually segmented into multiple slots, while MOVi-E results show cleaner object masks. Also, SAVi degenerates to stripe patterns on MOVi-D, but is able to produce meaningful masks on MOVi-E. Compared to STEVE, SlotDiffusion usually has more consistent tracking results and less object sharing issues, especially on large objects. We also note that the shown examples of SlotDiffusion on MOVi-D have FG-ARI scores higher than 80%, despite having low visual quality. This may indicate that FG-ARI metric has been saturated, and we should use better metrics such as mIoU (Karazija et al., 2021).

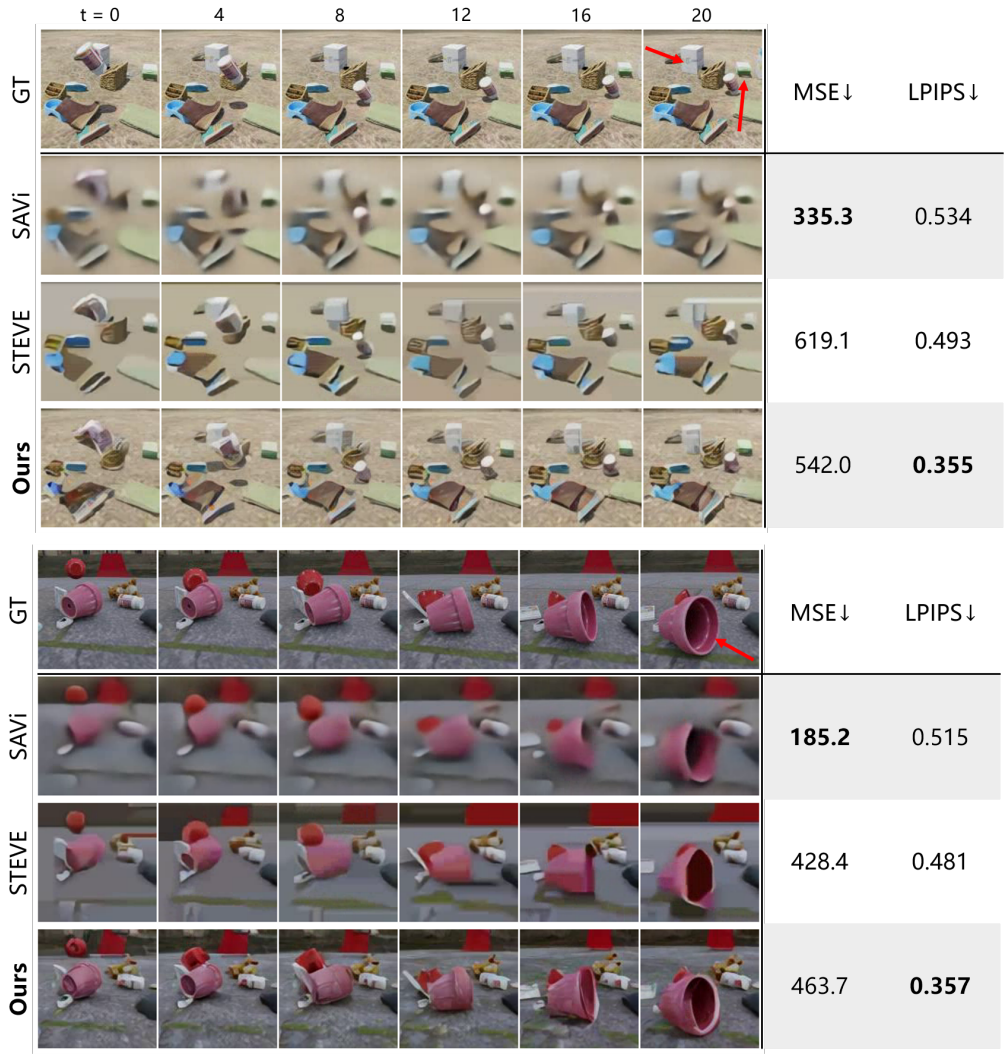


Figure 6: Qualitative results of video reconstructions on MOVi-E datasets.

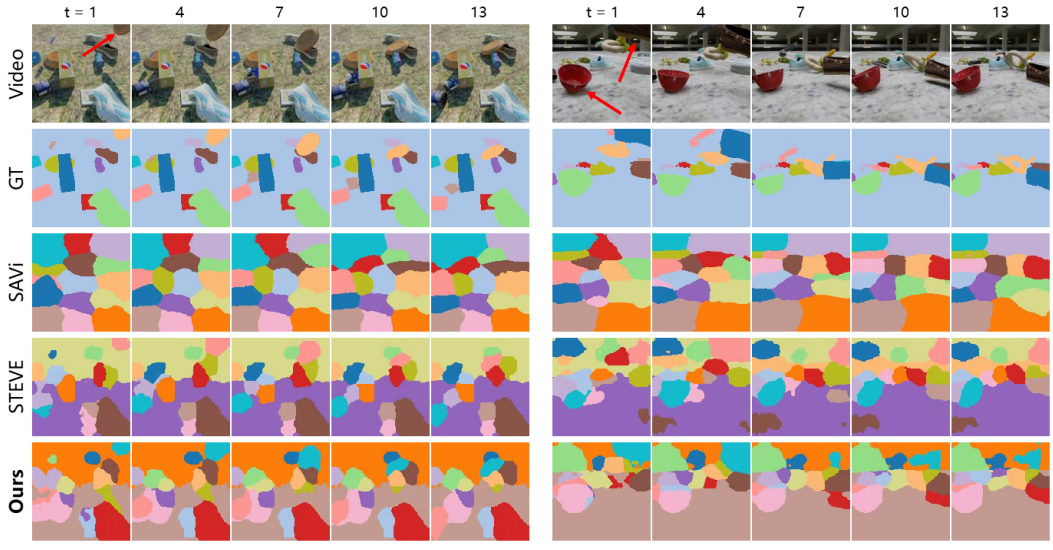


Figure 7: Video segmentation results on MOVi-D. The moving objects are highlighted with red arrows.

E.3 Ablation Study

We study the effect of the number of diffusion process steps T in our DM-based decoder. We plot both the reconstruction and segmentation results on two videos datasets in Figure 9. As expected,

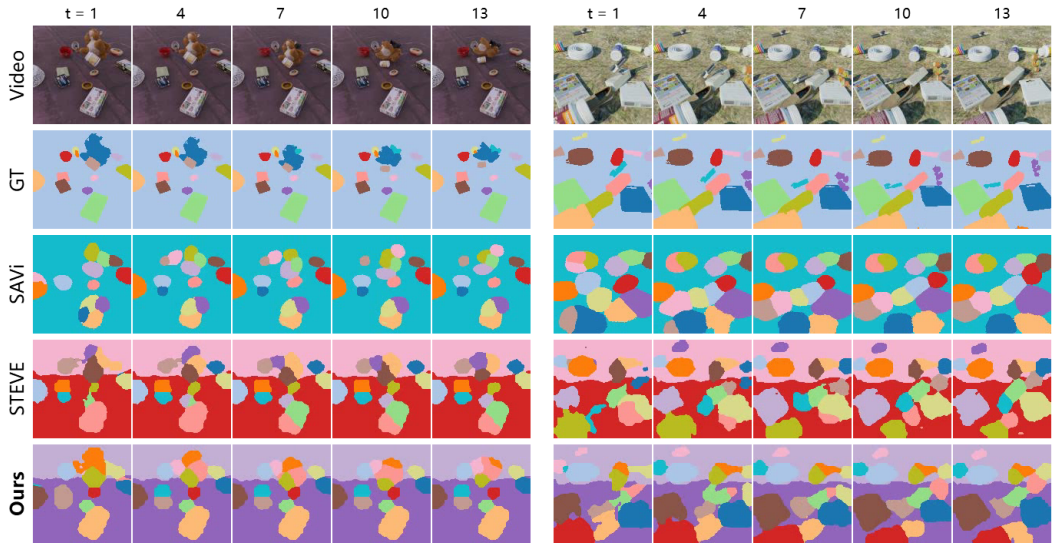


Figure 8: Video segmentation results on MOVi-E, where videos have small linear camera motion.

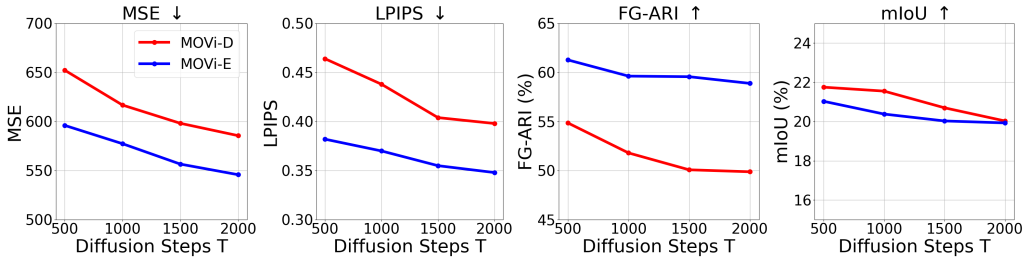


Figure 9: Ablation study on the number of diffusion steps T of SlotDiffusion. We show the reconstruction performance and segmentation results on both video datasets.

more denoising steps leads to better generation quality, thus lower MSE and LPIPS. Interestingly, smaller T results in better segmentation performance. This indicates that there is a trade-off between the reconstruction and segmentation quality of our model, and we select $T = 1000$ to strike a good balance. In the literature of self-supervised representation learning (He et al., 2022; Hua et al., 2022), it is believed that more difficult pretext tasks usually lead to better learned representations. Therefore, we hypothesize that a smaller T makes the pretext denoising task harder, resulting in better object-centric representations.

This observation aligns with one failure case in our experiments. Due to bad initialization, sometimes (1 out of 5 runs) SlotDiffusion degenerates to stripe pattern, where each slot captures a fixed region of the image instead of tracking an object. Surprisingly, the failed models usually have a much lower ($\sim 40\%$) reconstruction MSE compared to well-trained models. This raises a question about intrinsic problems of object-centric models that prevent them from doing good reconstructions.

E.4 Failed Attempts

Here, we record some failed model variants we tried.

Image-space diffusion model. At the early stage of this work, we adopt an image-space DM as the slot decoder without the VQ-VAE tokenizer. This works well with low resolution inputs (64×64). However, when we increase the resolution to 128×128, we observe color shifting artifacts which is a common issue among high resolution DMs (Saharia et al., 2022b;a). Switching to LDM-based decoder solves this problem, as the VQ-VAE decoder always map latent codes to images with natural color statistics. In addition, since LDM consumes less memory, we can use a larger U-Net as the denoiser, which further improves the performance.

Prediction target of the diffusion model. By default, our DM adopts the traditional noise prediction (ϵ) target. Recent works in DM propose two other formulations, namely, data prediction ($\mathbf{x}_0$) (Ramesh et al., 2022) and velocity prediction ($\mathbf{v}$) (Ho et al., 2022a), and claim to have better generation quality. We tried both targets in SlotDiffusion, but both model variants degenerate to the stripe pattern solution (similar to SAVi results in Figure 7) under multiple random seeds.

F LIMITATIONS AND FUTURE WORKS

Limitations. The goal of SlotDiffusion is to improve the generation quality of object-centric models. Currently, we only evaluate our model in terms of slot-to-image reconstruction, which is not a typical generation task. Also, although we improve the LPIPS by a sizeable margin, our MSE results are not very good. This may because MSE favors blurry results generated by baselines. Finally, as analyzed in Section E.2, the modeling capability of fine local details can still be improved.

Future Works. Recent work (Seitzer et al., 2022) has shown the power of contrastive pre-trained representations in unsupervised object discovery. We may add the contrastive loss to our framework to better improve the learned features. Also, we want to test our model's scene decomposition capacity on more video datasets, especially the real-world ones proposed by Singh et al. (2022). In addition, we plan to integrate SlotDiffusion with object-centric dynamics model (Wu et al., 2022) to tackle the video prediction task, where we can evaluate the performance gain brought by the improved slot decoder. Finally, to examine the quality of learned slots, we can apply our object features to downstream reasoning tasks such as VQA (Yi et al., 2019; Bear et al., 2021). Overall, we believe SlotDiffusion provides a better trade-off between generation quality and segmentation performance by introducing diffusion model to the field of object-centric learning, which we see as a promising direction.