Compositional Generative Models

For Generalizable Scene Generation and Understanding

Yanbo Wang

COMPOSITIONAL GENERATIVE MODELS

FOR GENERALIZABLE SCENE GENERATION AND UNDERSTANDING

COMPOSITIONAL GENERATIVE MODELS

FOR GENERALIZABLE SCENE GENERATION AND UNDERSTANDING

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology
by the authority of the Rector Magnificus,
prof. dr. ir. H. Bijl,
chair of the Board for Doctorates
to be defended publicly on
Thursday, 21 May 2026, 10:00

by

Yanbo WANG

This dissertation has been approved by the promotors.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Prof. dr. ir. G.J.T. Leus,	Delft University of Technology, *promotor*
Dr. ir. J.H.G. Dauwels,	Delft University of Technology, *promotor*
Independent members:	
Prof. dr. ir. M.J.T. Reinders,	Delft University of Technology
Prof. dr. ir. B. de Vries,	Eindhoven University of Technology
Dr. J.C. van Gemert,	Delft University of Technology
Dr. ir. Y. Chen,	Delft University of Technology
Prof. dr. ir. A.J. van der Veen,	Delft University of Technology, reserve member

Keywords: Compositional Generative Modeling, Object-Centric Generation, Scene Decomposition, Scene Understanding, Compositional Generalization

ISBN 978-94-6518-319-0

An electronic copy of this dissertation is available at
https://repository.tudelft.nl/.

We are all part of something much greater than us.

Contents

SUMMARY

Human intelligence is fundamentally compositional: it constructs new ideas by flexibly recombining known concepts, enabling generalization to entirely new tasks. We aim to develop intelligent systems with similar robust generalization capabilities. To that end, we develop compositional generative modeling frameworks and present three research thrusts that advance scene generation, decomposition, and understanding.

First, we introduce a hierarchical object-centric generative model that integrates latent-variable modeling with object-centric representation learning, enabling coherent multi-object scene generation and fine-grained object-level editing. This approach overcomes limitations of prior object-aware models by supporting flexible object morphology and significantly improving in-distribution generalization.

Second, we propose an unsupervised compositional image decomposition method that represents images as compositions of energy landscapes encoded by diffusion models. This enables the extraction of reusable global and local visual factors, such as shadows, expressions, and objects, and supports zero-shot compositional image generation by recombining these factors into novel configurations far outside the training distribution.

Third, we develop a compositional inverse generative modeling framework for scene understanding. By formulating inference as likelihood maximization over conditional generative model parameters, we show how composable diffusion models enable object discovery and multi-label classification in scenes substantially more complex than those seen during training, including generalization to images with more objects or new configurations. The framework also supports zero-shot category inference using pretrained generative models without additional training.

Overall, these contributions demonstrate that the incorporation of compositional structure into generative modeling yields interpretable, controllable, and significantly more generalizable intelligent systems. This thesis offers a step toward building intelligent agents with the flexible, systematic compositional imagination characteristic of human cognition.

SAMENVATTING

Menselijke intelligentie is fundamenteel compositioneel: zij construeert nieuwe ideeën door bekende concepten flexibel te combineren, wat generalisatie naar volledig nieuwe taken mogelijk maakt. Wij streven ernaar intelligente systemen te ontwikkelen met vergelijkbare robuuste generalisatiecapaciteiten. Daartoe ontwikkelen wij compositionele generatieve modelleringskaders en presenteren we drie onderzoekslijnen die generatie, decompositie en begrip van scènes bevorderen.

Ten eerste introduceren we een hiërarchisch object-gecentreerd generatief model dat latent-variabelemodellering integreert met object-gecentreerd representatieleren, waardoor coherente generatie van scènes met meerdere objecten en verfijnde objectniveaubewerking mogelijk wordt. Deze benadering overwint beperkingen van eerdere objectbewuste modellen door flexibele objectmorfologie te ondersteunen en de generalisatie binnen de distributie aanzienlijk te verbeteren.

Ten tweede stellen we een onbewaakte compositionele afbeeldingsdecompositiemethode voor die afbeeldingen representeert als composities van energielandschappen gecodeerd door diffusie modellen. Dit maakt de extractie mogelijk van herbruikbare globale en lokale visuele factoren, zoals schaduwen, gezichtsuitdrukkingen en objecten, en ondersteunt zero-shot compositionele afbeeldingsgeneratie door deze factoren te recombineren in nieuwe configuraties ver buiten de trainingsdistributie.

Ten derde ontwikkelen we een compositioneel invers generatief modelleringskader voor scènebegrip. Door inferentie te formuleren als waarschijnlijkheidsmaximalisatie over conditionele parameters van generatieve modellen, laten we zien hoe composeerbare diffusie modellen objectontdekking en multi-label classificatie mogelijk maken in scènes die aanzienlijk complexer zijn dan die uit de training, inclusief generalisatie naar afbeeldingen met meer objecten of nieuwe configuraties. Het kader ondersteunt ook zero-shot categorie-inferentie met behulp van vooraf getrainde generatieve modellen zonder extra training.

Al met al tonen deze bijdragen aan dat het opnemen van compositionele structuur in generatieve modellering leidt tot beter interpreteerbare, controleerbare en aanzienlijk generaliseerbare intelligente systemen. Deze thesis biedt een stap richting het bouwen van intelligente agenten met de flexibele, systematische compositionele verbeeldingskracht die kenmerkend is voor menselijke cognitie.

PREFACE

Atoms constitute molecules, molecules constitute matter and living beings, and all of them together constitute the universe. The physical world is fundamentally compositional and we, as conscious beings, are part of it. Since ancient times, humans have sought to understand the underlying processes by which the world is formed, or, in other words, "generated" from elementary building blocks.

Inspired by this long-standing curiosity, my research explores how artificial systems can learn to understand and generate real-world data in a similarly compositional way. Just as atoms combine to form molecules and molecules to form matter, intelligent systems can benefit from learning to represent, decompose, and recombine elements of their environment. This thesis investigates compositional generative modeling as a framework for building such systems, focusing on how structured representations can improve generalization, interpretability, and control in complex visual scenes.

This thesis was carried out between 2021 and 2025 in the Signal Processing Systems (SPS) group of EEMCS faculty at TU Delft. It summarizes years of investigation into compositional generative modeling, combining ideas from representation learning, probabilistic modeling, and optimization. Along the way, I encountered both conceptual and practical challenges, from grappling with abstract ideas of modularity and compositionality to engineering scalable models that could work in real-world scenes. These challenges, though sometimes daunting, made the research process deeply rewarding.

Yanbo Wang
Delft, September 2025

1

INTRODUCTION

The physical world is inherently compositional: environments around us are typically formed from a diverse set of objects, whose categories and relationships vary and evolve. Humans effortlessly perceive this compositional structure and can reason and imagine in novel environments that contain object combinations never encountered before. Inspired by this observation, we posit that incorporating such compositionality into the design of modern artificial intelligence systems is crucial for enabling robust generalization to out-of-distribution settings.

1

1.1. BACKGROUND AND MOTIVATION

Given basic visual, language, or conceptual knowledge, human intelligence can compositionally reuse them to construct increasingly novel and complex ideas [1–3]. This capacity enables effective generalization, such as the acquisition of new skills, reasoning in unfamiliar environments, and solving previously unseen tasks. A simple example is the human capacity to create novel works of art by abstractly combining objects, scenery, and style, often in ways that transcend everyday experience. A more extraordinary example is humanity's ability to build spaceships capable of traveling to the Moon by integrating vast amounts of physical and engineering knowledge, despite such tasks being entirely unprecedented. Equipping machines with this kind of compositional imagination, rooted in human intelligence, has long been a central goal of the artificial intelligence community in the pursuit of developing generalizable intelligent agents.

Empowered by Internet-scale datasets and rapidly evolving deep learning techniques, modern AI research has made remarkable progress towards building imaginative agents, particularly through recent advances in generative models [4–7]. By modeling the distribution of training data, these models can faithfully generate novel data points that resemble those seen during training, ranging from natural language to high-dimensional images and videos. For example, large language models (LLMs) support impressive text generation, reasoning, planning, and question answering [8–10], while visual generative models allow photorealistic image and video generation that captures long textual descriptions [11, 12]. Beyond generation, when conditioned on various practical constraints, these generative models also enable downstream capabilities such as scene understanding, decision making, and robotic manipulation, achieving excellent test-time performance in numerous applications[13–15].

Despite this progress, the way generative model–based intelligent agents represent and understand the world remains far from truly compositional or generalizable [16]. Without explicitly modeling the elementary composable components exhibited in the data, these systems typically rely on training monolithic models that learn implicit and abstract intermediate representations of the data [17–19]. Such representations are usually non-interpretable and difficult to be composed for generating more complex concepts. For example, in the visual domain, many generative models treat input images holistically, compressing each image into a single scene-level representation while ignoring the inherent object-centric structure. This modeling framework makes it difficult, if not impossible, to systematically compose objects from different scenes into a coherent new image. Moreover, these generative models are also typically trained under the assumption that the data distribution at inference time is the same as that of training. This assumption severely limits their capacity to generalize beyond narrowly defined distributions and makes them vulnerable to even slight distribution shifts [20]. For example, models trained on images containing a fixed number of objects are typically unable to generalize to images with more objects.

In practice, real-world environments are open-ended, constantly evolving, and naturally compositional. To adapt to such environments, collecting exhaustive training data to cover all potential situations is often prohibitively expensive or simply infeasible. Even at massive data scales, contemporary visual generative models often

fail at simple compositional image generation tasks, such as composing object pairs that are usually not present together. This suggests that the sheer scale of the data and parameters alone is insufficient to achieve the kind of compositional generalization that comes naturally to humans. To operate robustly under real-world conditions, intelligent systems must go beyond interpolation within a training distribution. They must be able to recombine known concepts to generate or reason about new ones and robustly handle novel configurations without requiring extensive retraining.

1.2. RESEARCH SCOPE AND CONTRIBUTIONS

To improve the generalizability of state-of-the-art generative models, this thesis focuses on developing compositional visual generative models that can compose factor-aware generative models to tackle complex visual tasks. Toward that end, the thesis centers on the following thrusts.

THRUST I: COMPOSITIONAL SCENE GENERATION

Scope. The first thrust investigates how modular inductive biases can promote compositional generative modeling to enable interpretable, controllable, and generalizable image generation. Conventional generative models typically learn a direct mapping from a random noise latent space to an image pixel space through monolithic neural networks [4, 6]. This mapping, however, cannot reflect how object components are formed, arranged, and related on the scene, leading to unexplainable, uncontrollable, and ungeneralizable generation of multi-object scenes. Until very recently some prior work [21, 22] proposed to build object-aware generative models that generate multi-object scenes component by component instead of solely pixel by pixel. However, these models are not flexible enough to capture image components of complex morphology due to the bounding-box representations used therein. Some other works [23–25] replace the bounding boxes with more flexible segmentation masks to represent object components, but the images generated by these approaches are quite blurry and inconsistent with the scene structures exhibited in the training set. This suggests that these models cannot faithfully generalize even in-distribution when training data contain delicate and subtle scene structures. We address these challenges by integrating object-centric representations encouraged through modular inductive biases with latent variable-generative models.

Contributions. We propose a compositional generative model that generates images hierarchically in an object-centric manner. Specifically, our proposed model first draws a scene-level representation from a random noise latent space, then maps this scene-level representation to multiple object-level representations, and finally renders an image by composing and decoding these object-level representations. We leverage the state-of-the-art object-centric learning approach Slot Attention [26] to iteratively attend objects on the scene and learn flexible representations of object components through modular inductive biases. We first illustrate how the integration of slot attention and latent variable models allows us to infer both global representation and object-centric representation simultaneously without any supervision. We further illustrate how the hierarchical combination of scene-level representations and

object-level representations ensures the coherency of generated scene structures and improves generated image quality significantly, demonstrating significantly better in-distribution generalization than baselines. Finally, we illustrate how our approach allows editing individual object components without affecting others, highlighting object-centric disentanglement ability.

THRUST II: COMPOSITIONAL SCENE DECOMPOSITION

Scope. The second thrust explores how to extract composable visual concepts with semantic meaning from images in an unsupervised manner to allow for generalizable zero-shot image generation by recombining the learned composable visual concepts at test time. Existing work [27] has demonstrated how to learn semantically meaningful latent variables through an information bottleneck in generative models. However, these latent variables are typically stacked in a single latent vector that models images holistically without explicitly encoding factor-wise information. Hence, reconfiguring a certain variable in the latent vector does not correspond to manipulating an individual visual factor, leaving the image generation procedure opaque. Another line of research [26, 28, 29] decomposes images with segmentation masks and represents objects with slot representations. Although achieving object-centric modeling, these approaches combine object components with rigid masks, leading to incoherent image generation when composing objects from different images. Meanwhile, slot representations struggle to model high-level relationships between factors, as well as multiple global factors that collectively affect the same image. We address these challenges by learning composable energy landscapes encoded in diffusion models. These energy landscapes can not only represent local factors like objects, but also capture global factors such as facial expressions, enabling compositional zero-shot image generation through energy composition.

Contributions. We propose an unsupervised image decomposition approach that can decompose input images into a set of reusable compositional components and recombine these components for novel image generation through diffusion models. Specifically, we first draw a connection between energy-based models and diffusion models by showing how representing an image as a composition of energy landscapes is equivalent to representing it as a composition of diffusion models conditioned on latent representations. We then learn the compositional image decomposition model by encoding each image into multiple latent vectors, taking the latent vectors as conditioning for a set of diffusion models, and composing diffusion models to reconstruct the input image. We illustrate how our proposed approach can faithfully decompose images into global factors such as shadows or facial expressions and local factors such as constituent objects. We further illustrate how the learned factors can be flexibly recomposed to generate images sharply different from those seen in training time, demonstrating a strong generalization ability.

THRUST III: COMPOSITIONAL SCENE UNDERSTANDING

Scope. The third thrust examines the possibilities of leveraging compositional generative models to tackle scene understanding tasks, especially focusing on generalizable inference of scene properties from images significantly more complex

than seen during training. Recent works [13, 14, 30, 31] have explored how expressive generative models, such as diffusion models, can be used in inference tasks. For example, generative classifiers [13] demonstrate promising results in generalizing to mild distribution shifts for classification problems. The major findings therein suggest that generative models can alleviate shortcut learning, i.e., overemphasizing features that are spuriously related to image labels compared to conventional discriminative models, enabling robust image classification. However, generative classifiers are typically designed for single-object classification, where test data differ only slightly from those seen during training. It remains elusive how to tackle more general scene understanding tasks, such as object discovery or multi-label classification in images containing multiple objects, through the lens of generative models, particularly when the scenes encountered at test time are substantially different from the training data. We address these challenges by modeling images compositionally with composable diffusion models and formulate scene understanding tasks as an inverse generative modeling problem.

Contributions. We propose a compositional inverse generative modeling framework that supports inferring scene properties from images more complex than those seen during training. Specifically, we formulate the inference of scene properties as finding conditional parameters of a visual generative model that best interpret an image. We learn these scene properties by maximizing the likelihood of an image with respect to conditional parameters. To enable inferring more scene properties, for example, more object locations than those seen during training, we further propose to build the visual generative model compositionally from smaller models over pieces of a scene. We illustrate how our proposed approach can accurately infer a set of objects in the image, enabling generalization to scenes with more objects than training images by fitting a larger set of conditional parameters for more generative models. We further illustrate how this approach can also allow multi-label classification tasks and enable generalization to new scenes as well. Finally, we illustrate how our approach can be directly applied to pre-trained generative models without requiring any additional training for zero-shot inference of object categories in images never seen during training.

RELATED PUBLICATIONS

1. **Yanbo Wang**, Justin Dauwels, Yilun Du, "Generalizable Scene Understanding through Compositional Energy Minimization", *In Review*, 2026.

2. **Yanbo Wang**, Justin Dauwels, Yilun Du, "Compositional Scene Understanding through Inverse Generative Modeling", *ICML*, 2025.

3. Jocelin Su*, Nan Liu*, **Yanbo Wang***, Joshua B. Tenenbaum, Yilun Du, "Compositional Image Decomposition with Diffusion Models", *ICML*, 2024. (* indicates equal contribution.)

4. **Yanbo Wang**, Letao Liu, Justin Dauwels, "Slot-VAE: Object-Centric Scene Generation with Slot Attention", *ICML*, 2023.

1.3. DISSERTATION OUTLINE

The rest of the thesis is organized as follows. Chapter 2 reviews several deep generative models that are used in the following chapters. Chapter 3 introduces a compositional generative model that composes object components through the hierarchical latent variable framework, enabling effective generalization for image generation and flexible object disentanglement. Chapter 4 presents a compositional image decomposition approach that decomposes images into visual concepts represented as diffusion models, enabling unsupervised discovery of reusable image factors that can be recomposed to generate images significantly different from training images. Chapter 5 discusses how scene understanding problems can be formulated as compositional inverse generative models, enabling inferring a larger number of objects or more complex scene properties than those seen during training. Chapter 6 provides a conclusion to the proposed compositional generative approaches, identifies limitations of these approaches, and discusses potential future research directions.

REFERENCES

[1] N. Chomsky. *Aspects of the Theory of Syntax*. Cambridge: The MIT Press, 1965. URL: http://www.amazon.com/Aspects-Theory-Syntax-Noam-Chomsky/dp/0262530074.

[2] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman. "Building machines that learn and think like people". In: *Behavioral and brain sciences* 40 (2017), e253.

[3] S. M. Frankland and J. D. Greene. "Concepts and compositionality: in search of the brain's language of thought". In: *Annual review of psychology* 71 (2020), pp. 273–303.

[4] D. P. Kingma and M. Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[5] D. P. Kingma and P. Dhariwal. "Glow: Generative flow with invertible 1x1 convolutions". In: *arXiv preprint arXiv:1807.03039* (2018).

[6] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*. Vol. 1. MIT Press, 2016.

[7] J. Ho, A. Jain, and P. Abbeel. "Denoising diffusion probabilistic models". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6840–6851.

[8] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *et al.* "Improving language understanding by generative pre-training". In: (2018).

[9] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.* "Language models are unsupervised multitask learners". In: *OpenAI blog* 1.8 (2019), p. 9.

[10] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.* "Language models are few-shot learners". In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[11] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. arXiv: 2112.10752 [cs.CV].

[12] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, *et al.* "Imagen video: High definition video generation with diffusion models". In: *arXiv preprint arXiv:2210.02303* (2022).

[13] A. C. Li, M. Prabhudesai, S. Duggal, E. Brown, and D. Pathak. "Your diffusion model is secretly a zero-shot classifier". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2206–2217.

[14] T. Amit, T. Shaharbany, E. Nachmani, and L. Wolf. “Segdiff: Image segmentation with diffusion probabilistic models”. In: *arXiv preprint arXiv:2112.00390* (2021).

[15] Y. Du, T. Lin, and I. Mordatch. “Model Based Planning with Energy Based Models”. In: *CoRL* (2019).

[16] N. Liu, S. Li, Y. Du, A. Torralba, and J. B. Tenenbaum. “Compositional Visual Generation with Composable Diffusion Models”. In: *arXiv preprint arXiv:2206.01714* (2022).

[17] R. Vedantam, I. Fischer, J. Huang, and K. Murphy. “Generative Models of Visually Grounded Imagination”. In: *ICLR*. 2018.

[18] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. “Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework”. In: *ICLR*. 2017.

[19] K. K. Singh, U. Ojha, and Y. J. Lee. “Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 6490–6499.

[20] Y. Du and L. Kaelbling. “Compositional generative modeling: A single model is not all you need”. In: *arXiv preprint arXiv:2402.01103* (2024).

[21] E. Crawford and J. Pineau. “Spatiial Invariant Unsupervised Object Detection with Convolutional Neural Networks”. In: *Thirty-Third AAAI Conference on Artificial Intelligence*. 2019.

[22] J. Jiang and S. Ahn. “Generative neurosymbolic machines”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 12572–12582.

[23] M. Engelcke, A. R. Kosiorek, O. P. Jones, and I. Posner. “Genesis: Generative scene inference and sampling with object-centric latent representations”. In: *arXiv preprint arXiv:1907.13052* (2019).

[24] M. Engelcke, O. Parker Jones, and I. Posner. “GENESIS-V2: Inferring Unordered Object Representations without Iterative Refinement”. In: *arXiv preprint arXiv:2104.09958* (2021).

[25] P. Emami, P. He, S. Ranka, and A. Rangarajan. “Slot Order Matters for Compositional Scene Understanding”. In: *arXiv preprint arXiv:2206.01370* (2022).

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

[27] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. “beta-vae: Learning basic visual concepts with a constrained variational framework”. In: *International conference on learning representations*. 2017.

[28] K. Greff, S. Van Steenkiste, and J. Schmidhuber. “Neural expectation maximization”. In: *Advances in Neural Information Processing Systems* 30 (2017).

[29] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner. “Monet: Unsupervised scene decomposition and representation”. In: *arXiv:1901.11390* (2019).

[30] P. Jaini, K. Clark, and R. Geirhos. “Intriguing properties of generative classifiers”. In: *arXiv preprint arXiv:2309.16779* (2023).

[31] K. Clark and P. Jaini. “Text-to-image diffusion models are zero shot classifiers”. In: *Advances in Neural Information Processing Systems* 36 (2024).

2

PRELIMINARIES

Recent years have seen significant progress in developing highly expressive generative models that can generate data resembling real-world high-dimensional observations. In this chapter, we review several deep generative modeling approaches that are used in the following chapters to facilitate understanding our proposed approaches, including variational auto-encoders, energy-based models and denoising diffusion probabilistic models. We introduce how these approaches can effectively model high-dimensional data distribution with feasible computational resources and generate novel data samples resembling training data. We also introduce how we can compose probabilistic distributions to build more complex distributions through product-of-experts, which can enable generalization to out-of-distribution data, as we will elaborate in the following chapters.

2.1. VARIATIONAL AUTO-ENCODERS

2.1.1. APPROACH OVERVIEW

2

Variational auto-encoder (VAE) [1, 2] assumes that the data generation process for continuous x proceeds in two steps: first, a latent variable is drawn $z \sim p(z)$, and then z is mapped to the data space via a probabilistic distribution $p(x|z)$. Under this assumption, the likelihood distribution $p(x|z)$ can be modeled by a parametric family $p_\theta(x|z)$ with learnable parameters θ, and similarly for the prior $p_\theta(z)$.

Training the generative model requires the posterior:

$$p_\theta(z|x) = \frac{p_\theta(z)p_\theta(x|z)}{p_\theta(x)}, \tag{2.1}$$

which is generally intractable due to the marginal likelihood $p_\theta(x)$:

$$p_\theta(x) = \int p_\theta(x|z)p_\theta(z)dz. \tag{2.2}$$

To circumvent this, VAEs use variational inference to approximate the intractable posterior $p_\theta(z|x)$ with a learnable surrogate $q_\phi(z|x)$ with parameters ϕ.

2.1.2. TRAINING OBJECTIVE

Instead of directly maximizing $\log p_\theta(x)$, VAEs optimize the evidence lower bound (ELBO) :

$$\log p_\theta(x) \geq \underbrace{\mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right]}_{\text{reconstruction term}} - \underbrace{D_{KL}\left[q_\phi(z|x)||p_\theta(z)\right]}_{\text{regularization term}}, \tag{2.3}$$

where $p_\theta(z)$ is typically chosen as the standard normal $\mathcal{N}(0, I)$, and $D_{\mathrm{KL}}(q||p)$ represents the Kullback-Leibler Divergence regularizing the approximate posterior $q_\phi(z|x)$ to be close to the prior $p_\theta(z)$. To enable learning expressive generative models in high-dimensional spaces, both the likelihood $p_\theta(x|z)$ and the posterior $q_\phi(z|x)$ are assumed to be Gaussian and parameterized with neural networks:

$$p_\theta(x|z) = \mathcal{N}\left(\mu_\theta(z), \sigma^2_\theta(z)\right), \quad q_\phi(z|x) = \mathcal{N}\left(\mu_\phi(x), \sigma^2_\phi(x)\right), \tag{2.4}$$

where $\mu_\theta(\cdot)$, $\sigma^2_\theta(\cdot)$, $\mu_\phi(\cdot)$, and $\sigma^2_\phi(\cdot)$ are differentiable functions implemented with neural networks appropriate for the data type. Instead of fitting a separate variational distribution per x, VAEs employ amortized inference that uses a single posterior network to approximate posteriors for all data points, enabling scaling to large datasets.

A major challenge in maximizing the ELBO with respect to both the variational parameters ϕ and the generative parameters θ is that the gradients cannot easily backpropagate through the sampling of $z \sim \mathcal{N}\left(\mu_\phi(x), \sigma^2_\phi(x)\right)$ [3]. To overcome this challenge, VAE proposes a reparameterization trick:

$$z = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon, \tag{2.5}$$

where $\epsilon \sim \mathcal{N}(0, I)$. This reparameterization separates the randomeness from the trainable parameters ϕ and allows gradients to flow through $\mu_\phi(\cdot)$ and $\sigma^2_\phi(\cdot)$. As a result, the ELBO can be rewritten as:

$$\mathbb{E}_{q_\phi(z|x)}\big[\log p_\theta(x|z)\big] - D_{KL}\big[q_\phi(z|x)||p_\theta(z)\big] \tag{2.6}$$

$$= \mathbb{E}_{p(\epsilon)}\big[\log p_\theta(x|(\mu_\phi(x) + \sigma_\phi(x) \odot \epsilon))\big] - D_{KL}\big[q_\phi(z|x)||p_\theta(z)\big] \tag{2.7}$$

$$= \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x|(\mu_\phi(x) + \sigma_\phi(x) \odot \epsilon_l)) - D_{KL}\big[q_\phi(z|x)||p_\theta(z)\big], \tag{2.8}$$

which is differentiable and can be optimized via standard backpropagation.

Once trained, a VAE can generate novel samples by sampling $z \sim p_\theta(z) = \mathcal{N}(0, I)$ and decoding $x \sim p_\theta(x|z)$. Conversely, the variational inference model $p_\phi(z|x)$ provides a latent representation of the input data. In this sense, VAE is similar to an autoencoder from the architecture perspective but differs in two key aspects: (1) VAE encodes each input into a distribution over z instead of a deterministic vector; (2) the training objective of VAE involves a probabilistic KL term instead of merely reconstruction. These differences enable a smooth latent space that supports interpolation and generation.

Although VAE has made significant contributions enabling end-to-end training of latent variable models parameterized with neural networks, it has some limitations. One limitation is that VAEs often produce blurry images for complex distributions because of Gaussian likelihood. Furthermore, the choice of prior $p(z)$ affects model expressiveness because standard Gaussian is simple but may limit flexibility. There is a large body of work that extends VAEs for a more powerful capacity. For example, NVAE [4] has explored hierarchical latent variable models trained under the variational autoencoder framework which demonstrates better sample quality than the original VAE. Another example is VQ-VAE [5] that proposes learning a discrete latent space and a learnable prior to circumvent issues of posterior collapse.

2.1.3. DISENTANGLEMENT PROPERTY

VAEs also exhibit a degree of disentanglement, where varying a latent dimension corresponds to changing a specific factor of variation in the data. For example, training on the CelebA human face image dataset [6] may allow one dimension to control a certain facial attribute such as hair color. To enhance this disentanglement property, β-VAE [7] introduces a hyperparameter $\beta \geq 0$ to weight the KL term :

$$\mathbb{E}_{q_\phi(z|x)}\big[\log p_\theta(x|z)\big] - \beta D_{KL}\big[q_\phi(z|x)||p_\theta(z)\big]. \tag{2.9}$$

When $\beta > 1$, $q_\phi(z|x)$ is constrained to be closer to $p_\theta(z) = \mathcal{N}(0, I)$, and thus the dimensions of z become less correlated and are encouraged to encode the individual and interpretable factor of variations. Despite its usefulness, both standard VAE and β-VAE still show incomplete disentanglement, i.e., some latent dimensions may remain entangled. Furthermore, they overlook useful data structure. For complex data like multi-object images, encoding the entire scene into a single latent vector entangles object features, limiting controllability and scene structure modeling.

In chapter 3, we explore object-level disentanglement for image data through the lens of VAE modeling. We focus on multi-object image distribution learning and illustrate how to incorporate object-centric structures into VAE modeling to enable better latent space interpolation for image generation, i.e., in-distribution generalization and object-centric controllable generation capabilities.

2.2. Energy-based Models

2.2.1. Approach Overview

Energy-based Models (EBMs) are a class of generative models that define a flexible probabilistic distribution over data x through an energy function $E_\theta(x)$:

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z_\theta}, \tag{2.10}$$

where $E_\theta(x)$ can be parameterized by any differentiable nonlinear function with parameters θ, and Z_θ denotes the partition function that normalizes the distribution:

$$Z_\theta = \int e^{-E_\theta(x)} dx. \tag{2.11}$$

The partition function is typically intractable due to the high-dimensional integral involved, which complicates the maximum likelihood training.

2.2.2. Training Objective

The log-likelihood can be written as:

$$\log p_\theta(x) = -E_\theta(x) - \log Z_\theta \tag{2.12}$$

where the second term, $\log Z_\theta$, cannot be evaluated directly. Nevertheless, its gradients with respect to θ can be calculated as [8]:

$$\nabla_\theta \log p_\theta(x) = -\nabla_\theta E_\theta(x) - \nabla_\theta \log Z_\theta \tag{2.13}$$

$$= -\nabla_\theta E_\theta(x) - \nabla_\theta \log \int e^{-E_\theta(x)} dx \tag{2.14}$$

$$= -\nabla_\theta E_\theta(x) + \frac{\int \nabla_\theta E_\theta(x) e^{-E_\theta(x)} dx}{\int e^{-E_\theta(x)} dx} \tag{2.15}$$

$$= -\nabla_\theta E_\theta(x) + \mathbb{E}_{p_\theta(x)} \nabla_\theta E_\theta(x). \tag{2.16}$$

Maximizing the log-likelihood therefore is equivalent to reducing the energy of training samples and increasing the energy value of samples drawn from the model. This procedure gradually shapes an energy landscape that implicitly defines the data distribution.

Estimating the gradient in Equation 2.16 requires sampling from model distribution Equation 2.10, which is non-trivial due to the intractable partition function. A large body of work has explored how to draw samples from EBMs efficiently, among

which the Langevin dynamics approach is widely used due to the improved mixing properties [9] compared to classical MCMC techniques such as Gibbs sampling [10]. In particular, Langevin dynamics generates data samples by exploring the energy landscape via stochastic gradient steps:

$$x^{t+1} \leftarrow x^t + \frac{\lambda}{2}\nabla_x \log p_\theta(x^t) + \omega^t, \omega^t \sim \mathcal{N}(0, \lambda) \tag{2.17}$$

$$\leftarrow x^t - \frac{\lambda}{2}\nabla_x E_\theta(x^t) + \omega^t, \omega^t \sim \mathcal{N}(0, \lambda), \tag{2.18}$$

where Equation 2.18 makes use of the fact that $\nabla_x \log Z_\theta = 0$. In the limit $\lambda \to 0$ and $t \to \infty$, Langevin dynamics samples exactly from the model distribution $p_\theta(x)$ under some regularity conditions.

Although Langevin dynamics mixes relatively fast, running long Markov chains to convergence remains expensive for gradient estimation in Equation 2.16 [11]. Several works such as Contrastive Divergence (CD) [12] and persistant CD [13] proposed truncated MCMC with running only a small number of MCMC steps from datapoint x initialization, which enables the maximum likelihood training practical but at the cost of introducing bias into distribution modeling. Alternative approaches such as score matching [14] and denoising score matching [15] bypass the need to sample from $p_\theta(x)$ through learning the score $\nabla_x \log p_\theta(x) = -\nabla E_\theta(x)$ with Fisher divergence as objective, inspired by the observation that knowing the score of $p_\theta(x)$ is enough to get samples from the distribution as suggested by Langevin dynamics. However, these approaches require computing second-order derivatives or introduce bias through artificially added noise. More recently, Annealed Score Matching [16] proposed to gradually corrupt data with increasing noise levels and learn the score function at each level. At sampling time, noise-to-data annealed Langevin process is run gradually at each noise-level. This approach is scalable and produces high-quality samples with rapid mixing.

2.2.3. ENERGY COMPOSITION

Beyond flexibility, EBMs have the attractive property that they can be easily combined to build more complex models. The product of EBMs actually corresponds to the sum of the corresponding energy functions [17, 18]:

$$p_\theta^1(x) p_\theta^2(x) \cdots p_\theta^N(x) \propto e^{-(E_\theta^1(x) + E_\theta^2(x) + \cdots + E_\theta^N(x))}. \tag{2.19}$$

Running Langevin dynamics with the summed energy yields samples from the product distribution.

In Chapter 4 and Chapter 5, we explore how we can compose distributions by summing energy landscapes as in Equation 2.19, enabling generalization to out-of-distribution data in various tasks.

2.3. DENOISING DIFFUSION PROBABILISTIC MODELS

Denoising diffusion probabilistic models (DDPMs) [19, 20] or diffusion models have found broad applications across a wide range of domains, owing to their exceptional

expressiveness and ability to capture complex, high-dimensional data distributions. In visual tasks, diffusion-based generative models have achieved state-of-the-art performance in image synthesis [21] and have been extended to video generation with impressive temporal coherence [22]. Beyond pure generation, diffusion models have also been adapted for discriminative tasks such as image classification [23–25] by leveraging their robust representation learning capabilities. Their flexibility further enables strong performance in dense prediction tasks, including semantic and medical image segmentation [26, 27]. More recently, diffusion models have also demonstrated promise in decision-making and control, where the generative formulation naturally supports planning, trajectory optimization, and policy learning [28–30]. These diverse applications illustrate the versatility of diffusion modeling as a general-purpose framework for both generative and discriminative tasks.

2.3.1. APPROACH OVERVIEW

Diffusion models learn to generate data by reversing a diffusion process. Inspired by non-equilibrium thermodynamics, diffusion models define a forward diffusion process that gradually destroys the structure in the data by adding Gaussian noise, and a learned reverse diffusion process that reconstructs data from noise [20]. Specifically, diffusion models define the generative model as the marginal of a reverse process:

$$p_\theta(x_0) = \int p_\theta(x_{0:T})\, dx_{1:T}, \tag{2.20}$$

where $x_1, ..., x_T$ are noise-corrupted versions of the data $x_0 \sim q(x_0)$ and the joint distribution $p_\theta(x_{0:T})$ takes a factorized form:

$$p_\theta(x_{0:T}) = p_\theta(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t). \tag{2.21}$$

To facilitate data sampling, $p_\theta(x_T)$ is typically chosen as $\mathcal{N}(0, I)$, while $p_\theta(x_{t-1}|x_t)$ is modeled with te Gaussian distribution $\mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$, where mean and variance are parameterized with neural networks.

The diffusion process that produces the corrupted augmentation data $x_1, ..., x_T$ is typically described with a Markov chain:

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}), \quad q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t I), \tag{2.22}$$

where β_t are noise schedules that can be either fixed or made learnable. It has been shown in [31] that when β_t is small enough, the reverse process and the diffusion process have the same functional form. That justifies why the factors in Equation 2.21 are defined in a Gaussian form.

2.3.2. TRAINING OBJECTIVE

Maximizing directly $\log p_\theta(x)$ is intractable, and a surrogate to maximum likelihood training can be ELBO maximization:

$$\mathbb{E}\big[\log p_\theta(x_0)\big] \geq \mathbb{E}_q\Big[\log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)}\Big] = \mathbb{E}_q\Big[\log p_\theta(x_T) - \sum_{t\geq 1} \log \frac{p_\theta(x_{t-1}|x_t)}{q(x_t|x_{t-1})}\Big]. \tag{2.23}$$

Next, we further simplify Equation 2.23 for training in practice.

Note that for the forward process in Equation 2.22, we do not have to add noise step by step to arrive at the time step t. We can sample x_t in one step by noticing that:

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I), \tag{2.24}$$

where $\alpha_t = 1-\beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t}\alpha_s$. Using this property, we can rewrite $q(x_t|x_{t-1})$ as:

$$q(x_t|x_{t-1}) = q(x_t|x_{t-1}, x_0) = \frac{q(x_{t-1}|x_t, x_0)q(x_t|x_0)}{q(x_{t-1}|x_0)}, \tag{2.25}$$

where $q(x_{t-1}|x_t, x_0) = \mathcal{N}(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I)$, with $\tilde{\mu}_t(x_t, x_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t$ and $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$. Substituting Equation 2.25 into Equation 2.23, we have the following objective:

$$\mathbb{E}_q\big[\underbrace{-D_{KL}(q(x_T|x_0)||p_\theta(x_T))}_{L_T} - \sum_{t>1}\underbrace{D_{KL}(q(x_{t-1}|x_t, x_0)||p_\theta(x_{t-1}|x_t))}_{L_{t-1}} + \log\underbrace{p_\theta(x_0|x_1)}_{L_0}\big], \tag{2.26}$$

where all KL terms are tractable. Assuming β_t are fixed and distribution $q(x_t|x_0)$ does not have learnable parameters, L_T is a constant. Furthermore, by assuming $\Sigma_\theta(x_t, t) = \sigma_t^2 I$, we have:

$$L_{t-1} = \mathbb{E}_q\big[\frac{1}{2\sigma_t^2}||\tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)||^2\big] + C. \tag{2.27}$$

Recalling that $q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I)$, we can represent $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{(1-\bar{\alpha}_t)}\epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$. Then we can rewrite Equation 2.27 as

$$L_{t-1} = \mathbb{E}_{x_0,\epsilon}\big[\frac{1}{2\sigma_t^2}\Big|\Big|\frac{1}{\sqrt{\alpha_t}}\Big(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon\Big) - \mu_\theta(x_t, t)\Big|\Big|^2\big] + C. \tag{2.28}$$

From Equation 2.28, it is straightforward to parameterize the following:

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)), \tag{2.29}$$

where ϵ_θ is a neural network that predicts the noise added to x_0 at the time step t. As a result, Equation 2.28 can be rewritten as:

$$\mathbb{E}_{x_0,\epsilon}\Big[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\Big|\Big|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\Big|\Big|^2\Big] + C. \tag{2.30}$$

Empirically, [20] finds that the following simplified version of objective for training diffusion models yields better sample quality:

$$L = \mathbb{E}_{t\sim Uniform(0,..,T), x_0\sim q(x_0), \epsilon\sim\mathcal{N}(0,I)}\Big[\Big|\Big|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\Big|\Big|^2\Big]. \tag{2.31}$$

Once trained, diffusion models generate samples by first drawing $x_T \sim \mathcal{N}(0, I)$, and then running the reverse denoising procedure until $t = 0$. Specifically, given:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}\Big(x_{t-1}; \frac{1}{\sqrt{\alpha}_t}\Big(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\Big), \sigma_t^2 I\Big), \tag{2.32}$$

the reverse process predicts the sample at time step $t-1$ as:

$$x_{t-1} = \frac{1}{\sqrt{\alpha}_t}\Big(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\Big) + \sigma_t w, \quad w \sim \mathcal{N}(0, I), \tag{2.33}$$

which suggests that the sampling procedure corresponds to gradually removing predicted noise from noisy data until obtaining a synthetic sample x_0. Although this reverse process traditionally requires hundreds to thousands of steps, later works such as DDIM [32] introduced deterministic and accelerated sampling, enabling high-quality samples with far fewer iterations.

2.3.3. CONNECTION TO EBMS

In chapter 4 and chapter 5, we explore how we can compose diffusion models for scene decomposition and understanding, enabling generalization to out-of-distribution data at the test time. We draw a close connection between energy-based models and diffusion models to motivate the way of composing diffusion models. In particular, the denoising step of the diffusion model in Equation 2.33 resembles a single step of the Langevin dynamic sampling of EBMs in Equation 2.18. That is, the noise prediction $\epsilon_\theta(x_t, t)$ in the diffusion model is equivalent to the estimated score $E_\theta(x)$ in EBMs. Similar to products of experts in Equation 2.19 where each expert is modeled with EBMs, we can also model the experts with diffusion models. To that end, we can learn a set of diffusion models $\epsilon_\theta^1(x_t, t), \epsilon_\theta^2(x_t, t), ..., \epsilon_\theta^N(x_t, t)$ to represent distributions $p_\theta^1(x), p_\theta^2(x), p_\theta^N(x)$, respectively. To sample from product-of-experts $\prod_{n=1}^N p_\theta^n(x)$, we can sum the noise prediction of each expert, analogously to the sum of scores in EBMs, and then denoise at time step $t-1$ as follows:

$$x_{t-1} = \frac{1}{\sqrt{\alpha}_t}\Big(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\sum_{n=1}^N \epsilon_\theta^n(x_t, t)\Big) + \sigma_t w, \quad w \sim \mathcal{N}(0, I). \tag{2.34}$$

The connection between EBMs and diffusion models can also be bridged with score-based models [16]. In [20], it is shown that diffusion models are equivalent to score-based models and that the diffusion process corresponds to learning the score field at multiple noise levels. This connection further supports that noise prediction in diffusion models is essentially the learned score field of energy landscapes that can be composed together to build more complex distributions.

2.4. SUMMARY

In this chapter, we provide an overview of mathematical foundations of variational auto-encoder, energy-based models, and denoising diffusion probabilistic models.

We introduce distribution formulations, learning objectives, training challenges and solutions, as well as methods for sampling from the learned models. Furthermore, we also illustrate how generative models can be composed to construct more complex probabilistic models. In the following chapters, we propose compositional generative approaches based on the models introduced in this chapter to enable effective generalization for several vision tasks.

REFERENCES

[1] D. P. Kingma and M. Welling. “Auto-encoding variational bayes”. In: *arXiv preprint arXiv:1312.6114* (2013).

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

[3] J. Paisley, D. Blei, and M. Jordan. “Variational Bayesian inference with stochastic search”. In: *arXiv preprint arXiv:1206.6430* (2012).

[4] A. Vahdat and J. Kautz. “NVAE: A deep hierarchical variational autoencoder”. In: *Advances in neural information processing systems* 33 (2020), pp. 19667–19679.

[5] A. Van Den Oord, O. Vinyals, *et al.* “Neural discrete representation learning”. In: *Advances in neural information processing systems* 30 (2017).

[6] T. Karras, T. Aila, S. Laine, and J. Lehtinen. “Progressive growing of gans for improved quality, stability, and variation”. In: *arXiv preprint arXiv:1710.10196* (2017).

[7] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. “beta-vae: Learning basic visual concepts with a constrained variational framework”. In: *International conference on learning representations*. 2017.

[8] L. Younes. “On the convergence of Markovian stochastic algorithms with rapidly decreasing ergodicity rates”. In: *Stochastics: An International Journal of Probability and Stochastic Processes* 65.3-4 (1999), pp. 177–228.

[9] M. Welling and Y. W. Teh. “Bayesian learning via stochastic gradient Langevin dynamics”. In: *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 2011, pp. 681–688.

[10] G. E. Hinton. “Training products of experts by minimizing contrastive divergence”. In: *Training* 14.8 (2006).

[11] Y. Song and D. P. Kingma. “How to train your energy-based models”. In: *arXiv preprint arXiv:2101.03288* (2021).

[12] G. E. Hinton. “Training products of experts by minimizing contrastive divergence”. In: *Neural computation* 14.8 (2002), pp. 1771–1800.

[13] T. Tieleman. “Training restricted Boltzmann machines using approximations to the likelihood gradient”. In: *Proceedings of the 25th international conference on Machine learning*. ACM. 2008, pp. 1064–1071.

[14] A. Hyvärinen and P. Dayan. “Estimation of non-normalized statistical models by score matching.” In: *Journal of Machine Learning Research* 6.4 (2005).

[15] P. Vincent. “A connection between score matching and denoising autoencoders”. In: *Neural computation* 23.7 (2011), pp. 1661–1674.

[16] Y. Song and S. Ermon. “Generative Modeling by Estimating Gradients of the Data Distribution”. In: *Advances in Neural Information Processing Systems*. 2019, pp. 11895–11907.

[17] G. E. Hinton. “Products of experts”. In: *International Conference on Artificial Neural Networks* (1999).

[18] Y. Du and L. P. Kaelbling. “Position: Compositional generative modeling: A single model is not all you need”. In: *Forty-first International Conference on Machine Learning*. 2024.

[19] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. “Deep Unsupervised Learning using Nonequilibrium Thermodynamics”. In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by F. Bach and D. Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 2256–2265. URL: https://proceedings.mlr.press/v37/sohl-dickstein15.html.

[20] J. Ho, A. Jain, and P. Abbeel. “Denoising diffusion probabilistic models”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6840–6851.

[21] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. arXiv: 2112.10752 [cs.CV].

[22] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, *et al.* “Imagen video: High definition video generation with diffusion models”. In: *arXiv preprint arXiv:2210.02303* (2022).

[23] A. C. Li, M. Prabhudesai, S. Duggal, E. Brown, and D. Pathak. “Your diffusion model is secretly a zero-shot classifier”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2206–2217.

[24] A. C. Li, A. Kumar, and D. Pathak. “Generative Classifiers Avoid Shortcut Solutions”. In: *ICML 2024 Workshop on Structured Probabilistic Inference {\&} Generative Modeling*. 2024.

[25] K. Clark and P. Jaini. “Text-to-image diffusion models are zero shot classifiers”. In: *Advances in Neural Information Processing Systems* 36 (2024).

[26] T. Amit, T. Shaharbany, E. Nachmani, and L. Wolf. “Segdiff: Image segmentation with diffusion probabilistic models”. In: *arXiv preprint arXiv:2112.00390* (2021).

[27] E. A. Brempong, S. Kornblith, T. Chen, N. Parmar, M. Minderer, and M. Norouzi. “Denoising pretraining for semantic segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 4175–4186.

[28] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine. “Planning with diffusion for flexible behavior synthesis”. In: *arXiv preprint arXiv:2205.09991* (2022).

[29] Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. Tenenbaum, D. Schuurmans, and P. Abbeel. “Learning universal policies via text-guided video generation”. In: *Advances in neural information processing systems* 36 (2023), pp. 9156–9172.

[30] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. “Diffusion policy: Visuomotor policy learning via action diffusion”. In: *The International Journal of Robotics Research* 44.10-11 (2025), pp. 1684–1704.

[31] W. Feller. “On the Theory of Stochastic Processes, with Particular Reference to Applications”. In: *First Berkeley Symposium on Mathematical Statistics and Probability*. 1949, pp. 403–432.

[32] J. Song, C. Meng, and S. Ermon. “Denoising diffusion implicit models”. In: *arXiv preprint arXiv:2010.02502* (2020).

3

Compositional Scene Generation

In this chapter, we explore how image generation can be approached in a compositional manner by leveraging recent advances in object-centric learning, to enable compositional generalization and structure-aware generation. Specifically, we propose a compositional generative model that integrates slot attention with the hierarchical latent variable framework for object-centric structured scene generation. For each image, the model simultaneously infers a global scene representation to capture high-level scene structure and object-centric slot representations to embed individual object components. During generation, slot representations are generated from the global scene representation to ensure coherent scene structures. Our extensive evaluation of scene generation ability indicates that our proposed approach outperforms slot representation-based generative baselines in terms of sample quality and scene structure accuracy, demonstrating better in-distribution generalization.

This chapter is based on the paper published in Proceedings of the 40th International Conference on Machine Learning, PMLR 202:36020-36035, (2023) [1].

3.1. INTRODUCTION

3.1.1. BACKGROUND AND MOTIVATION

Human intelligence is capable of visually segmenting objects out of natural scenes, implicitly learning abstract object concepts, and creatively imagining novel scenes [2, 3]. Equipping machines with such capabilities in an unsupervised way has been a desideratum for a long time [4–7], since this can facilitate intelligent agents understanding scenes, reasoning about object relationships, and performing tasks efficiently [8–15]. To that end, most recent models resort to the variational autoencoder (VAE) framework [16, 17] for the purpose of joint object-centric representation inference and image generation. Depending on how to model the compositionality of images, existing works can be roughly categorized as spatial attention-based generative models and scene-mixture-based generative models.

Spatial attention-based generative models infer object-centric representations by extracting a bounding box for each individual object [18–22]. Such bounding boxes explicitly represent the position and size of the object components, enabling interpretable object manipulation. However, this type of model was pointed out to struggle in segmenting objects with extensively varied scales because the size of objects is, to some extent, presumed [23, 24]. Moreover, rectangular bounding boxes are also not flexible enough to model image components of complex morphology [20]. In contrast, scene-mixture generative models decompose a visual scene into image-sized components (also known as slots), and infer slot representations corresponding to individual objects [23, 25–27]. Such models segment objects with masks and are flexible enough to capture complex object components. Recent advances in scene-mixture models have shown remarkable object segmentation performance [23, 27]. However, although the design of such models advocates autoregressive priors for the purpose of generating coherent scenes, they are still unable to model object relationships in highly structured images, and the generated samples are very blurry.

In this chapter, we propose an object-centric generative model termed Slot-VAE that integrates slot attention with the hierarchical VAE framework for joint slot representation inference and structured image generation. In the proposed model, object-centric representation inference is achieved with the slot attention module [28]. Although slot attention has shown very impressive unsupervised segmentation performance, it is a deterministic module without the ability to generate novel scenes. If we naïvely combine slot attention with vanilla VAE for multi-object image generation, the generated images would be unreasonable because slots are completely independent and the scene structure (e.g., object relationships) is ignored. To overcome this issue, we adopt a two-layer hierarchical VAE model, which provides both global scene representations that capture the scene structure and object-centric slot representations that characterize individual objects. Slot representations are generated from global scene representations during the generation stage to ensure a coherent scene structure. During training, in addition to learning from global scene representations, slot representations are also regularized by an independent prior to encourage object-centric disentanglement. Furthermore, the variational framework and independent prior also bring slot attention the attribute-level disentanglement.

Evaluating on several multi-object datasets, we show that Slot-VAE outperforms baselines in terms of sample quality and scene structure learning.

3.1.2. CHAPTER CONTRIBUTIONS

The contributions of this chapter are as follows. First, we introduce a generative model that embeds slot attention into the principled latent variable modeling framework for novel scene generation. Second, we incorporate a hierarchical latent variable model to learn both scene-level and object-centric representations. Third, we empower the slot attention baseline with the object attribute-level disentanglement ability. Lastly, extensive experimental results suggest that our proposed method outperforms the state-of-the-art methods in terms of sample quality and scene structure accuracy.

3.2. RELATED WORK

Object-Centric Generative Modeling. Compositional image modeling approaches [19, 20, 25, 26, 28–36] typically incorporate object locality as inductive bias or exploit simple decoder networks as reconstruction bottlenecks [37] to achieve object-centric disentanglement. However, these approaches, unlike ours, cannot generate coherent novel scenes. GENESIS and GENESIS-V2 [27] [23] adopt autoregressive prior for coherent scene generation, but unlike ours, they lack the scene-level representation learning ability and generate blurry samples. GNM [22] and similarly GSGN [38] resort to a hierarchical VAE model for both distributed and symbolic representations learning, but the bounding box representations therein prevent them from modeling complex objects or backgrounds, unlike ours where more flexible slot representations are used. SRI [24] learns slot representations and scene-level representations, but it has to sequentially infer object representations due to the assumed autoregressive posterior. In contrast, our approach poses an independent prior on slot representations allowing parallel inference. Besides, our approach trains the model without the need to learn a fixed object order, but SRI requires specialized auxiliary loss for object order learning so as to train the model.

GANs, EBMs, and Diffusion Models for Compositional Generation. GANs-based methods [39–43] are able to map independent random noise vectors to individual object components on images allowing object-level controllability, but these models lack an inference process and thus cannot learn object-centric representations for downstream tasks, such as editing a specific object without affecting others in a given image, unlike ours. Additionally, these GANs models share common unstable training issues. Energy-based models have also been used for compositional image generation as demonstrated by [44–47]. Nevertheless, the energy-based models therein are trained with MCMC sampling, which is computationally intensive and difficult to scale to high-dimensional data, resulting in blurry generated samples. Similar to GAN-based methods, EBM-based compositional generative models cannot infer object masks or global image representations effectively for downstream tasks. Furthermore, these approaches model compositional generation through the conditioning conjunction, lacking the ability to generate images with subtle structures unconditionally. More recently, [48] proposed generating images by composing powerful diffusion models.

While effective, similar to EBM-based compositional approaches, these methods also lack an inference procedure to explicitly learn object-centric representations and therefore cannot support fine-grained image editing. Moreover, their approach relies on text-conditioned generation and cannot generate images unconditionally, in contrast to our method.

3.3. COMPOSITIONAL SCENE GENERATION WITH OBJECT-CENTRIC REPRESENTATIONS

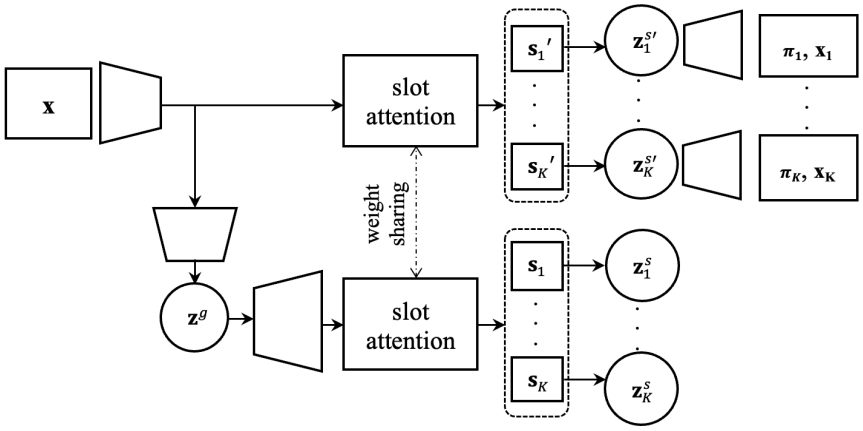

Figure 3.1: Slot-VAE overview. The image $\mathbf{x}$ is passed through a CNN module. The obtained image features go through two paths in parallel. On the first path, the obtained image features are input into a slot attention module to learn slot representations $\{\mathbf{s}'_k\}_{k=1}^K$. From slots $\{\mathbf{s}'_k\}_{k=1}^K$, latent vectors $\{\mathbf{z}_k^{s'}\}_{k=1}^K$ are inferred. Then, a shared decoder decodes the individual object latent vector $\{\mathbf{z}_k^{s'}\}_{k=1}^K$ into object masks $\boldsymbol{\pi}_{1:K}$ and object components $\mathbf{x}_{1:K}$. By combining $\mathbf{x}_{1:K}$ with $\boldsymbol{\pi}_{1:K}$, the input $\mathbf{x}$ is reconstructed. On the second path, the obtained image features is encoded into a global latent vector $\mathbf{z}^g$. From $\mathbf{z}^g$, a feature map is built and fed into a slot attention module to generate slot representations $\{\mathbf{s}_k\}_{k=1}^K$. From $\{\mathbf{s}_k\}_{k=1}^K$, latent vectors $\{\mathbf{z}_k^s\}_{k=1}^K$ are inferred. The two paths use the same slot attention module and share weights and initialization values, and it requires $\{\mathbf{z}_k^{s'}\}_{k=1}^K$ and $\{\mathbf{z}_k^s\}_{k=1}^K$ to be as close as possible during training measured with KL divergence.

In this section, we introduce our compositional generative model, Slot-VAE. We first illustrate how we can model the generation of object representations from global scene representations and the subsequent generation of images from these object representations by using hierarchical VAE. We then illustrate how the slot attention module can be leveraged for object-centric representation extraction during inference procedure. Finally, we illustrate the training objective as well as training techniques for Slot-VAE. We outline the overview of Slot-VAE in Fig. 3.1.

3.3.1. OBJECT-CENTRIC GENERATIVE MODELING

For an image $\mathbf{x} \in [0,1]^{H\times W\times C}$ with height H, width W and C channels, we postulate a two-layer hierarchical latent model for the potential image generation process. Specifically, the first-layer latent vector $\mathbf{z}^g \in \mathbb{R}^{L\times 1}$ captures the global structure in the image, for the purpose of modeling relationships among objects. Generated from $\mathbf{z}^g$, the second-layer latent vectors $\{\mathbf{z}^s_k \in \mathbb{R}^{D\times 1}\}_{k=1}^K$ represent each individual object in the image, with the goal of incorporating object-centric slot representations. These slot representations $\mathbf{z}^s_{1:K}$ are assumed to be conditionally independent given $\mathbf{z}^g$. Finally, with $\mathbf{z}^s_{1:K}$, an image $\mathbf{x}$ can be rendered with a decoder. Mathematically, the complete object-centric compositional generative model can be written as:

$$p_\theta(\mathbf{x}) = \iint p_\theta(\mathbf{x} \mid \mathbf{z}^s_{1:K}) p_\theta(\mathbf{z}^s_{1:K} \mid \mathbf{z}^g) p_\theta(\mathbf{z}^g) d\mathbf{z}^s_{1:K} d\mathbf{z}^g. \tag{3.1}$$

The global latent vector $\mathbf{z}^g$ serves as an information bottleneck to extract high-level information (e.g., object appearance, positions, and relations) for whole image reconstruction. $\mathbf{z}^g$ is similar to the latent vector in VAE but not exactly the same. The difference is that in VAE the latent vector is directly decoded to an image, while in Slot-VAE $\mathbf{z}^g$ is used to generate slot representations $\mathbf{z}^s_{1:K}$. For the prior of $\mathbf{z}^g$, we can choose a powerful StructDRAW prior [22] or a simple Normal distribution depending on image complexity.

Slot representations $\mathbf{z}^s_{1:K}$, in contrast to $\mathbf{z}^g$, ideally embed information of individual object components and totally ignore object relationships. Such object-centric representations explicitly model the compositional structure of images, enable compositional generation, and make the generation process interpretable. To generate $\mathbf{z}^s_{1:K}$ from $\mathbf{z}^g$, we first construct a feature map $\mathbf{f} \in \mathbb{R}^{H\times W\times D}$ from $\mathbf{z}^g$ and then feed $\mathbf{f}$ to a slot attention module [28] to obtain slot representations $\{\mathbf{s}_k \in \mathbb{R}^{D\times 1}\}_{k=1}^K$. Since slot attention is a deterministic module, an additional MLP is needed to map deterministic $\mathbf{s}_{1:K}$ to probabilistic latent vectors $\mathbf{z}^s_{1:K}$. Assuming $\mathbf{z}^s_{1:K}$ to be Gaussian and conditionally independent given $\mathbf{z}_g$, we have:

$$p_\theta(\mathbf{z}^s_{1:K} \mid \mathbf{z}^g) = \prod_{k=1}^K p_\theta(\mathbf{z}^s_k \mid \mathbf{z}^g). \tag{3.2}$$

The use of the slot attention module for object-centric latent vector generation sets the proposed Slot-VAE apart from GNM [22] where bounding box extraction is adopted. Such a difference brings the following key benefits. First, slot-based models have been shown to be more flexible in modeling objects with complex morphology compared to the spatial attention module [20]. Second, the dimension of the feature map $\mathbf{f}$ in GNM fundamentally limits the maximum number of components in an image to be $H \times W$. Once a GNM model is trained, it can infer $H \times W$ objects at most. In contrast, the slot attention module can successfully generalize to infer more object components even though it only saw K object components during training. Comes with these benefits a key challenge to Slot-VAE: there is no fixed order for the slot attention outputs. Since slot attention maps an input into a set (of slots), for the same input image, multiple runs may give the same set of slot representations but

with different orders. This is because slot attention employs random initialization for slots to achieve slot permutation symmetry. However, such randomness makes learning a hierarchical latent variable model extremely challenging, which we will explain in detail in Section 3.3 and contribute to solving it.

With $\mathbf{z}^s_{1:K}$, rendering an image $\mathbf{x}$ is as follows. First, from $\mathbf{z}^s_{1:K}$ (or $\mathbf{z}^{s'}_{1:K}$ in Fig. 3.1), K sub-images $\{\mathbf{x}_k \in [0,1]^{H\times W\times C}\}_{k=1}^K$ are rendered, each of which has the same dimension as $\mathbf{x}$ and ideally contains only one object. Meanwhile, this process also produces K object masks $\boldsymbol{\pi}_{1:K} \in [0,1]^{H\times W}$ corresponding to each $\mathbf{x}_k$. Then the image $\mathbf{x}$ is obtained by combining $\mathbf{x}_{1:K}$ with masks $\boldsymbol{\pi}_{1:K}$. Pixel-wisely, the likelihood can be written as

$$p_\theta(\mathbf{x}_{i,j} \mid \mathbf{z}^s_{1:K}) = \mathcal{N}\Big(\big(\sum_{k=1}^K \pi_{i,j,k}(\mathbf{z}^s_{1:K})\mu_{i,j,k}(\mathbf{z}^s_k)\big), \sigma_x^2\Big), \tag{3.3}$$

where (i,j) is the pixel coordinate, σ_x is the standard deviation with a fixed value, and $\pi_{i,j,k}(\cdot)$ and $\mu_{i,j,k}(\cdot)$ are nonlinear functions mapping from latent vectors to masks $\boldsymbol{\pi}_k$ and mean values of $\mathbf{x}_k$ at the pixel (i,j). These nonlinear functions are parameterized by neural networks with learnable parameters θ, and the implementation details are provided in the appendix. In equation 3.3, $\pi_{i,j,k}$ serves as a mixing probability, so it is constrained by $\sum_{k=1}^K \pi_{i,j,k} = 1, \forall(i,j)$.

In summary, to generate a novel scene, we first draw a random sample from the prior distribution of the global latent vector $\mathbf{z}^g$, from which a feature map $\mathbf{f}$ is built. Then, object-centric latent vectors $\mathbf{z}^s_{1:K}$ are generated using the slot attention module with the feature map $\mathbf{f}$ as input. Finally, the object components $\mathbf{x}_{1:K}$ and the corresponding masks $\boldsymbol{\pi}_{1:K}$ are generated from $\mathbf{z}^s_{1:K}$ with parallel decoders, and a novel scene is rendered by combining $\mathbf{x}_{1:K}$ with $\boldsymbol{\pi}_{1:K}$.

3.3.2. VARIATIONAL INFERENCE MODELING

Considering that the true posterior is intractable, we approximate the posterior with:

$$p_\theta(\mathbf{z}^g, \mathbf{z}^s_{1:K} \mid \mathbf{x}) \approx q_\phi(\mathbf{z}^g \mid \mathbf{x}) q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x}), \tag{3.4}$$

wherein the global latent posterior $q_\phi(\mathbf{z}^g \mid \mathbf{x})$ is modeled by an autoregressive model or Gaussian distribution depending on StructDRAW prior or Gaussian prior is used [22].

We further assume the factorization $q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x}) = \prod_{k=1}^K q_\phi(\mathbf{z}^s_k \mid \mathbf{x})$. Such conditional independence assumption on the posterior distribution of slot representations enables the inference of individual $\mathbf{z}^s_k$ to be performed in parallel, which avoids sequential inference like in GENESIS. We adopt slot attention [28] followed by an MLP to infer $\mathbf{z}^s_{1:K}$, which is detailed as follows.

CNN for feature extraction. Instead of directly working in the pixel domain, the slot representation inference starts from passing the input image $\mathbf{x}$ through a CNN backbone to extract a feature map $\mathbf{f}_x = f_{enc}(\mathbf{x}) \in \mathbb{R}^{H\times W\times D}$, where the CNN backbone is augmented with positional embeddings.

Slot attention for component discovery. To discover object components, the feature map $\mathbf{f}_x$ is first flattened into vectors $\mathbf{f}_{input} \in \mathbb{R}^{(H\times W)\times D}$. Then, $\mathbf{f}_{input}$ is mapped to K object slots $\mathbf{s}_{1:K}$ with a slot attention module.

MLP for latent vector inference. From slots $\mathbf{s}_{1:K}$, we would like to infer the latent variables $\mathbf{z}^s_{1:K}$. We assume that the approximate posterior distribution of each individual slot $q_\phi(\mathbf{z}^s_k \mid \mathbf{x})$ to be Gaussian. Hence, inferring $\mathbf{z}^s_k$ is equivalent to infer Gaussian parameters $\{(\mu^s_k, \sigma^s_k)\}_{k=1}^K$. To that end, we use an MLP shared across objects mapping from slots to Gaussian means and variances: $(\mu^s_k, \sigma^s_k) := \text{MLP}(\mathbf{s}_k)$.

3.3.3. TRAINING OBJECTIVE

Given the above generative and inference model, the evidence lower bound (ELBO) can be derived as follows:

$$\begin{aligned}\mathscr{L}(\mathbf{x};\theta,\phi) = \mathbb{E}_{q_\phi(\mathbf{z}^s_{1:K}|\mathbf{x})}\big[\log p_\theta(\mathbf{x} \mid \mathbf{z}^s_{1:K})\big] \\ - D_{\text{KL}}\big[q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z}^s_{1:K} \mid \mathbf{z}^g)\big] \\ - D_{\text{KL}}\big[q_\phi(\mathbf{z}^g \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z}^g)\big]\end{aligned} \tag{3.5}$$

where $D_{\text{KL}}(q||p)$ is the Kullback-Leibler Divergence.

Slot Order Matching in KL. Observing the second term on the RHS of equation 3.5, we can identify a key challenge for the calculation of this KL divergence: since the slots given by slot attention come with no fixed order, how can we determine the correspondence between $\mathbf{z}^s_{1:K}$ inferred from input $\mathbf{x}$ (which is denoted $\mathbf{z}^{s'}_{1:K}$ in Fig. 3.1) and $\mathbf{z}^s_{1:K}$ generated from $\mathbf{z}^g$? This challenge does not appear in GNM because the spatial attention module therein provides a fixed order for each object component, making the calculation of KL divergence in GNM possible. To address such a challenge in Slot-VAE, we propose to implement $q_\phi(\mathbf{z}^s_k \mid \mathbf{x})$ and $p_\theta(\mathbf{z}^s_k \mid \mathbf{z}^g)$ with a shared slot attention module. That is to say, as shown in Fig. 3.1, the two slot attention modules share parameters. Meanwhile, slots $\mathbf{s}^{'}_k$ and $\mathbf{s}_k$ in Fig. 3.1 share initialization values. Intuitively, such an architecture design encourages the feature map $\mathbf{f}$ generated from $\mathbf{z}_g$ to be consistent with the feature map $\mathbf{f}_x$ encoded from the input $\mathbf{x}$. With similar inputs and the same random initialization values, we can expect the output of the two slot attention modules to stay close to each other. As a result, the order of $\mathbf{s}_k$ (or $\mathbf{z}^s_k$) can have a good chance of aligning well with that of $\mathbf{s}^{'}_k$ (or $\mathbf{z}^{s'}_k$) in Fig. 3.1, allowing for the calculation of $D_{\text{KL}}\big[q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z}^s_{1:K} \mid \mathbf{z}^g)\big]$. We will empirically demonstrate the efficacy of such an architectural inductive bias for slot order matching in Section 4.

Furthermore, since $p_\theta(\mathbf{z}^s_{1:K} \mid \mathbf{z}^g)$ in the second term of equation 3.5 is learned from the posterior distribution $p_\theta(\mathbf{z}_g \mid \mathbf{x})$, it does not provide explicit prior information to guide the learning of the posterior distribution $q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x})$ during training. To explicitly provide guidance to the learning of $q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x})$, the following auxiliary loss could be incorporated:

$$\mathscr{L}_{aux} = -D_{\text{KL}}\big[q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x}) \,||\, \prod_{k=1}^{K} \mathcal{N}(\mathbf{0}, \mathbf{1})\big], \tag{3.6}$$

where an independent normal prior constrains $\mathbf{z}^s_{1:K}$ to be independent of each other. As a result, such independence encourages each slot representation to capture only a single object leading to object-centric disentanglement. Meanwhile, attribute-level

disentanglement within an object can also be achieved due to diagonal variance of the normal prior, which we will show in experiments.

Combining the ELBO derived in equation 3.5 and the auxiliary loss introduced in equation 3.6, the overall objective function is the following:

$$\tilde{\mathscr{L}} = \mathscr{L} + \mathscr{L}_{aux}. \tag{3.7}$$

For effective training, we also introduce hyperparameters to balance the reconstruction loss and KL terms [49, 50].

3.4. EXPERIMENTS

The experiments are to evaluate: i) image decomposition performance, ii) sample quality and structure accuracy of generated samples, iii) and disentanglement performance.

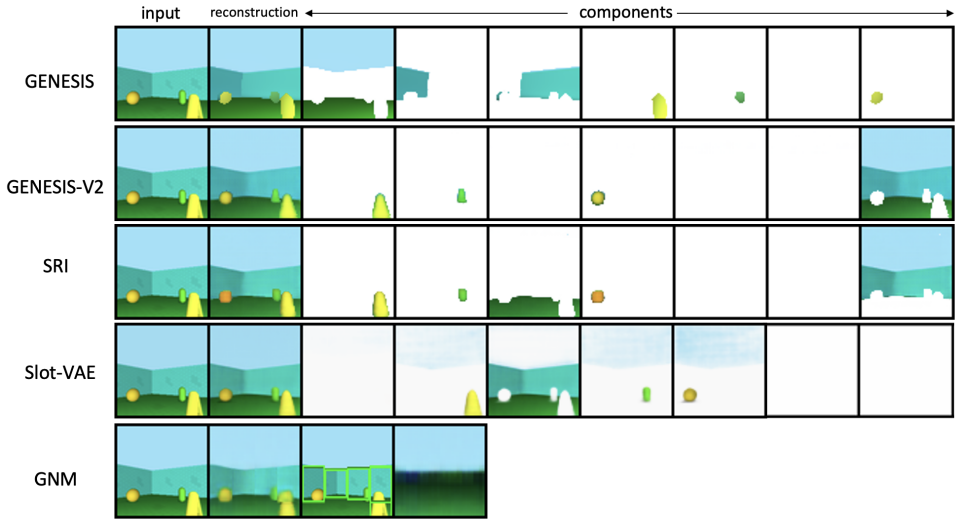

Figure 3.2: Image decompostion and reconstruction performance on the ObjectsRoom dataset.

Dataset. The experiments involve three datasets including *ObjectRoom* [51], *ShapeStacks* [52] and *Arrow Room*[22]. The datasets *ObjectRoom* and *ShapeStacks* are commonly used by previous works to test object-centric inference and generation, while *Arrow Room* is less considered because this dataset is highly structured and its probabilistic density is hard to model. In *Arrow Room*, there is always an arrow shape object in the front of the scene and three objects in the back. Two of the three objects in the back have the same shape, while a third one has a unique shape. The arrow in the front always points to the object with a unique shape in the back.

Baselines. We compare Slot-VAE against state-of-the-art object-centric generative models including GENESIS, GENESIS-V2, SRI and GNM. In these baselines, GNM is based on the spatial attention model (i.e., bounding box representations)

with hierarchical generation process, while GENESIS, GENESIS-V2 and SRI are scene-mixture models (i.e., slot representations) that assume an autoregressive prior. Some of the baseline models already released their trained models for *ObjectRoom*, *ShapeStacks* or *Arrow Room*. We do not retrain them and directly use their weights for comparison. For these of baseline models without trained models on some datasets, we train them with the official code.

3.4.1. SCENE DECOMPOSITION AND RECOMBINATION

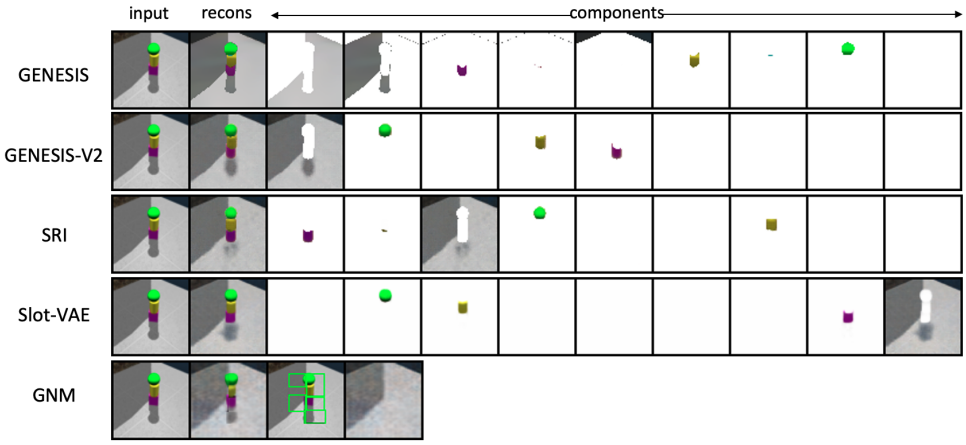

Figure 3.3: Image decompostion and reconstruction performance on the ShapeStacks dataset.

Decomposition and Reconstruction Performance. We illustrate the input, reconstruction and decomposed object components of Slot-VAE and baselines in Fig. 3.2 - 3.4. Note that GNM infers bounding box representations instead of slot representations. So in the figures, GNM has only two components, one for the foreground with bounding boxes and another for the background.

As shown in Fig. 3.2, for the *ObjectRoom* dataset that comes with simple object shapes and complex background components, scene mixture models GENESIS, GENESIS-V2, SRI and Slot-VAE achieve comparable decomposition and reconstruction performance. The only difference is that some of them capture the background with one slot while others use multiple slots. In contrast, GNM fails to segment objects correctly. It segments the scene into stripes containing parts of objects and parts of the background, and a single object is segmented into multiple bounding boxes. As a result, the reconstructed images of GNM show rectangular artifacts and objects are blurred. This is not surprising because with the use of grid sampling and bounding box representations, spatial-attention generative models like GNM struggle with modelling objects that have complex morphology. In Fig. 3.3, we observe similar results for the *ShapeStacks* dataset, where GENESIS, GENESIS-V2, SRI and Slot-VAE decompose and reconstruct the image reasonably well while GNM again tries to model one single object with multiple bounding boxes. Failing to learn correct

obeject-centric representations, GNM will also suffer during the generation stage as will be presented below. For the *Arrow Room* dataset that has simple object shapes but complex scene structures in Fig. 3.4, we can see all models successfully segment objects out of the scene and reconstruct the input image. However, GENESIS-V2 and SRI learn object representations that severely involve part of the background. Such representations will make the generated image samples very blurry, as will be shown below. We conjecture this is because the *Arrow Room* dataset has too strong object position relationships, and GENESIS-V2 and SRI (based on GENESIS-V2) do not have enough capacity and have to choose simple ways to segment images. In summary, Slot-VAE achieves either better or comparable segmentation and reconstruction performance in comparison to baselines. Additional decomposition results of Slot-VAE can be found in the Appendix.

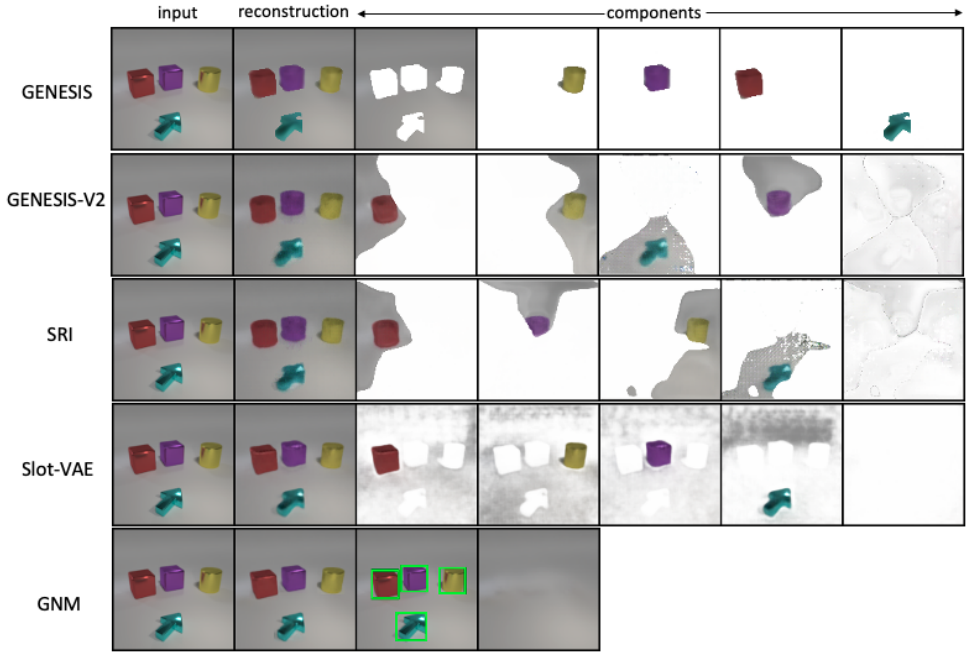

Figure 3.4: Image decompostion and reconstruction performance on the Arrow Room dataset.

3.4.2. SCENE GENERATION AND MANIPULATION

Generation Performance. We show random samples generated by Slot-VAE and baseline models in Fig. 3.5. It can be seen Slot-VAE generates the sharpest samples that highly resemble all the datasets. For *ObjectRoom*, samples generated by GNM show stripe artifacts due to its inaccurate object-centric representations captured by bounding boxes as discussed above. The sample quality of SRI is better than that of GENESIS and GENESIS-V2, but not as good as the proposed Slot-VAE. This can be reflected by the sharpness of object edges in the images. One can more easily identify

object shapes (e.g., balls and triangles) with Slot-VAE compared to baselines. For *ShapeStacks*, GNM again shows its limitation where it generates one individual object component with several parts. For example, a cube is represented by two small parts with completely different colors. Only SRI and Slot-VAE generate reasonable samples reflecting the scene structure of the *ShapeStacks* dataset (i.e., one object is stacked on another), while the sample quality of Slot-VAE is better in terms of sharp object edges. For *Arrow Room*, the most structured dataset, we find samples generated by GENESIS, GENESIS-V2 and SRI are very blurry and seldom show the underlying true scene structure (i.e., the arrow in the front always points to the object with a unique shape in the back). Both arrow directions or object shapes are not properly learned. This indicates that the autoregressive prior adopted in GENESIS, GENESIS-V2 and SRI is not strong enough to capture the complex scene structure in *Arrow Room*. In contrast, GNM and Slot-VAE, both exploiting hierarchical model to capture scene structure, generate very coherent and high-quality samples on the *Arrow Room* dataset. The reason why GNM works better on *Arrow Room* in comparison to *ObjectRoom* and *ShapeStacks* is that object shapes are simple in *Arrow Room*. In summary, Slot-VAE outperforms baselines in terms of sample quality and scene structure learning. Additional random generation results of Slot-VAE can be found in the Appendix.

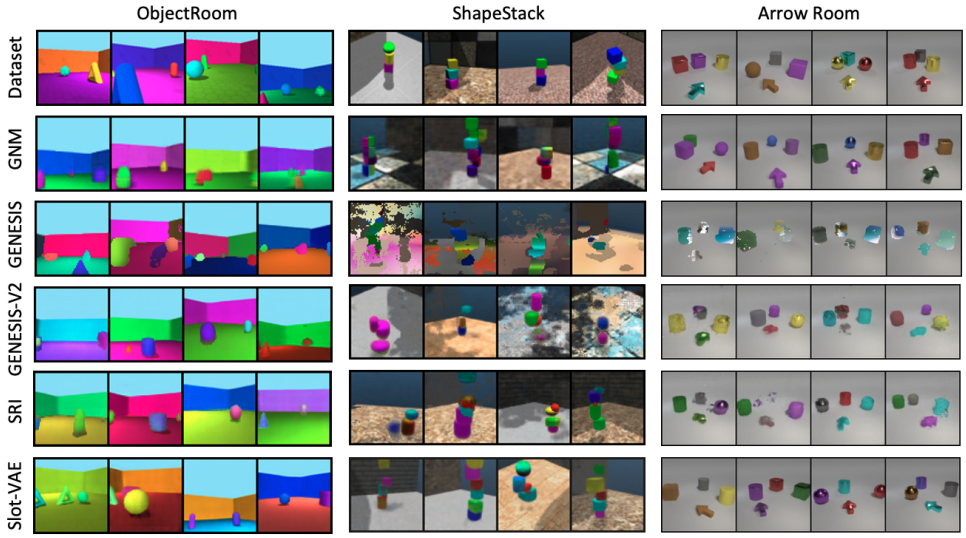

Figure 3.5: Datasets and generation examples of Slot-VAE and baselines.

Scene Manipulation. We elaborate on controllable scene generation to highlight the disentanglement performance of Slot-VAE. In Fig. 3.6, in each row we vary a certain dimension of the object-centric latent vector corresponding to the ball object while keeping other object-centric latent vectors unchanged. As is shown, only attributes of the ball are changed in each row, and all other objects remain unaffected. Such object-level disentanglement is very useful for image editing and compositional generation. Besides object-level disentanglement, attributes-level disentanglement also

Table 3.1: ARI-FG (↑) for Slot-VAE and Baselines on ObjectsRoom and ShapeStacks. Mean and standard deviation of ARI with three runs are presented. Scores labelled with * are from original works [37] and [24].

Model	ObjectsRoom	ShapeStacks
GNM	0.63* ± 0.00	0.37* ± 0.07
GENESIS	0.63* ± 0.03	0.70* ± 0.05
GENESIS-V2	0.84* ± 0.01	0.81* ± 0.00
SRI	0.83* ± 0.02	0.78* ± 0.02
Slot-VAE (ours)	0.79 ± 0.01	0.80 ± 0.01

Table 3.2: Fréchet Inception Distances (FID ↓) and Structure Accuracy (S-Acc ↑) for Slot-VAE and Baselines. Mean and standard deviation of FID with three runs are presented. Scores labelled with * are from original works [37] and [24].

	ObjectsRoom	ShapeStacks	Arrow Room	
Model	FID	FID	FID	S-Acc
GNM	51.6*±5	49.3*±2	11.2±2	0.97
GENESIS	62.8*±3	186.8*±18	173.8±13	0.11
GENESIS-V2	52.6*±3	112.7*±3	111.8±5	0.20
SRI	48.4*±4	70.4*±3	123.3±2	0.18
Slot-VAE (ours)	**34.9±1**	**50.0 ± 1**	**60.3±1**	**0.94**

naturally appears in Slot-VAE due to the adopted probabilistic framework. As shown in Fig. 3.6, when we vary dimension 1, the texture of the ball changes; when we vary dimension 2, the color of the ball changes; when we vary dimension 3, the size of the ball changes. Although some dimensions (e.g., dim 4) entangle color and position a little, this can be further improved with existing attribute-level disentanglement techniques like β-VAE [53] or β-TCVAE [54], which is out of the scope of this paper. In the proposed Slot-VAE, attribute-level disentanglement is a by-product brought by the VAE framework. By contrast, the original deterministic slot attention module comes with no obvious attribute-level disentanglement as analyzed in [35].

3.4.3. QUANTITATIVE COMPARISON

We report the Adjusted Rand Index (ARI) [55] score, Frechet Inception Distance (FID) [56] score and scene structure accuracy (S-Acc) [22] score to quantitatively evaluate the decomposition performance, sample quality, and scene structure accuracy. Since the *Arrow Room* dataset comes with no ground truth masks, the ARI score on this dataset is not calculated. As shown in Table 3.1, slot-VAE achieves comparable ARI scores to baselines. For the FID score, the calculation involves 10000 real and generated samples. Table 3.2 reflects non-trivial FID score improvement by Slot-VAE

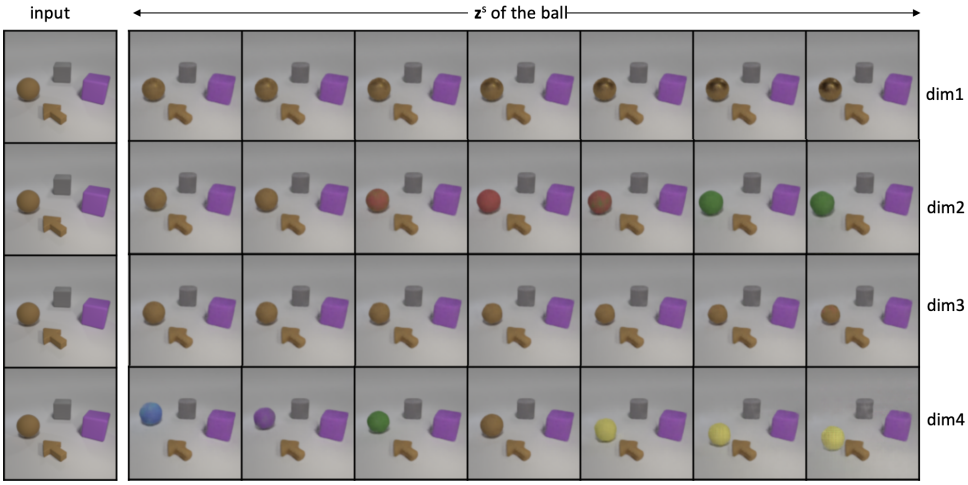

Figure 3.6: Slot-VAE latent traversal on *Arrow Room*. Each row only varies a certain dimension of $\mathbf{z}^s$ corresponding to the ball object.

against slot-representation baselines, highlighting the sample quality of Slot-VAE. Although the FID score of GNM on *ObjetsRoom* and *ShapeStacks* seems quite good, it should be emphasized that the generated images are unrealistic (i.e., generated objects are composed of multiple rectangular parts) due to inaccurate object representation learning as analyzed in the qualitative comparison results. For the S-Acc score, we manually classified 100 generated images per model, and calculated the ratio of successful images that correctly reflect scene structure. The datasets *ObjetsRoom* and *ShapeStacks* have relatively less clearly defined structures, which may result in difficulty in deciding if generated images truly reflect scene structures. To reduce subjective decisions, we mainly evaluate S-Acc of Slot-VAE and baseline models on the *Arrow Room* dataset because this dataset has a clearly defined structure: the arrow object should always point to the object with a unique shape in the back. Slot-VAE achieves the best S-Acc score among all the slot representation-based models (GENESIS, GENESIS-V2 and SRI), as is shown in Table 3.2.

3.4.4. ABLATION STUDY

We further conduct experiments to demonstrate the efficacy of the proposed architectural design in Fig. 3.1. Specifically, we aim to answer the following questions: (1) whether slot attention is necessary for generating slot representations from the global representation and (2) whether slot attention weight sharing and initialization value sharing are necessary for slot order matching. To that end, we evaluated the FID score and S-Acc score of several Slot-VAE variants.

To answer question (1), we investigate two approaches that could be used as alternatives to slot attention to generating slot representations $\{\mathbf{s}_k\}_{k=1}^K$ from the global representation $\mathbf{z}^g$. The first approach (termed as Slot-VAE-MLP) is by using an MLP to

Table 3.3: FID (↓) score and S-Acc (↑) score of Slot-VAE and variants on the *Arrow Room* dataset.

Model	FID	S-Acc
Slot-VAE	**60.3±1**	**0.94**
Slot-VAE-MLP	289±9	0.00
Slot-VAE-Transformer	182.1±3	0.03
Slot-VAE-W/O-WS	215.5±2	0.00
Slot-VAE-W/O-IVS	142.1±3	0.05

directly map the $\mathbf{z}^g$ to $\{\mathbf{s}_k\}_{k=1}^K$. Although this approach is straightforward, it cannot work well intuitively. Specifically, an MLP learns a deterministic mapping that always outputs slots $\{\mathbf{s}_k\}_{k=1}^K$ with a fixed order for a given global latent vector, whereas the slots $\{\mathbf{s}'_k\}_{k=1}^K$ that are directly inferred from the input image with slot attention come with a random order. As a result, the order of $\{\mathbf{z}^s_k\}_{k=1}^K$ and that of $\{\mathbf{z}^{s'}_k\}_{k=1}^K$ can rarely match each other, leading to fluctuating KL divergence $D_{\mathrm{KL}}\big[q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z}^s_{1:K} \mid \mathbf{z}^g)\big]$ between slot prior and slot posterior and hence diverged training. This can be reflected by the very high FID score and low S-Acc score in Table 3.3. The second approach (termed as Slot-VAE-Transformer) is by using a transformer to map the global vector $\mathbf{z}^g$ and random initialization values of slots $\{\mathbf{s}_k\}_{k=1}^K$ shared with $\{\mathbf{s}'_k\}_{k=1}^K$ to slot representations. In this approach, slots generated by the transformer is permutation invariant due to random initialization, which addresses the fixed slot order issue in Slot-VAE-MLP. Intuitively, with shared initialized values, slots $\{\mathbf{s}_k\}_{k=1}^K$ generated from $\mathbf{z}^g$ and slots $\{\mathbf{s}'_k\}_{k=1}^K$ inferred from the input image could have a good chance to match each other. Indeed, with this approach, our model matches the orders of the slots well. However, the generated slots turn out not so good in the sense that their corresponding decoded object components are very blurry. As a result, Slot-VAE-Transformer also has a very high FID score and a low S-Acc score. In contrast, Slot-VAE outperforms Slot-VAE-MLP and Slot-VAE-Transformer significantly, which demonstrates the effectiveness of slot attention for generating slot representations from the global representation.

To answer question (2), we trained a variant of Slot-VAE (termed as Slot-VAE-W/O-WS) without the weight sharing strategy in Fig. 3.1. In this case, the two slot attention modules update their weights respectively with no common initialization values. Without weight sharing, we anticipate that the KL divergence $D_{\mathrm{KL}}\big[q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z}^s_{1:K} \mid \mathbf{z}^g)\big]$ could be large because the learned slot representations of the two attention modules can be quite different, which may result in unrealistic generation samples. This is demonstrated by the experimental results in Table 3.3. We also trained another variant of Slot-VAE (termed as Slot-VAE-W/O-IVS) with weight sharing between the two slot attention modules but without initialization value sharing. Without initialization value sharing, the order of slots $\{\mathbf{s}_k\}_{k=1}^K$ generated from $\mathbf{z}^g$ and the order of slots $\{\mathbf{s}'_k\}_{k=1}^K$ inferred from the input image cannot match each other very well. As a result, the KL divergence $D_{\mathrm{KL}}\big[q_\phi(\mathbf{z}^s_{1:K} \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z}^s_{1:K} \mid \mathbf{z}^g)\big]$ can not

be properly calculated, and generated samples cannot reflect the dataset structure as quantitatively shown in Table 3.3.

In summary, we empirically find that the slot attention module for generating slot representations from the global representation, weight sharing and initialization value sharing between the two attention modules improve the generation performance significantly.

3.5. SUMMARY

Conclusions. We propose an object-centric generative model, Slot-VAE, that integrates the slot attention module with a hierarchical VAE model for joint object-centric representation inference and scene structure modelling. The proposed model can discover object components in an unsupervised way and generate novel scenes controllable at both the object and attribute level. This approach leads to interpretable image generation, resembling how human intelligence draws a picture in a component by component way. Experiment results show that Slot-VAE achieves better sampling quality and scene structure accuracy compared to slot representation-based generative baselines.

Limitations. One limitation of Slot-VAE is that the adopted slot attention module requires simple decoders like SBD [57] to serve as a reconstruction bottleneck to decompose objects, which, however, may not scale to complex real-world scenes. This can be improved by using a transformer decoder [32] or diffusion model-based decoder [58], which we leave for future work. Another limitation is that the slot-attention module predefines a fixed number of objects in the scene, leading to possible oversplitting of the scene when the actual number of objects is smaller than the prefefined number. For example, some blank slots capture parts of the background or parts of a complex object, which makes slot representations inaccurate. A potential remedy is to introduce additional sparsity prior to enforce sparsity on slot activations, which encourages slots to be silent unless they contribute meaningfully to reconstruction.

Social Impact. The proposed Slot-VAE model shows no negative social impacts in its current form since the evaluation is carried out on synthetic datasets at this stage. However, with improved slot representation learning modules available in the future, our model has the potential to be applied to generate more sophisticated and realistic scenes. In that case, misuse should be avoided for malicious purposes. Proper use of the proposed model can actually benefit practical applications like artwork generation, scene understanding, and dataset augmentation, to name just a few.

3.6. SUPPLEMENTARY

In this appendix, we present additional details on compositional scene generation. We illustrate additional qualitative results in Section 3.6.1, model details in Section 3.6.3, and training details in Section 3.6.4.

3.6.1. ADDITIONAL QUALITATIVE RESULTS

We show additional scene decomposition and scene generation results of Slot-VAE on *ObjectsRoom ShapeStacks* and *Arrow Room* in Fig.3.7 - Fig. 3.12. As we can see, Slot-VAE can effectively decompose images into meaningful components across all datasets, demonstrating a strong object-centric representation learning capability. Especially for the ShapeStacks dataset where multiple objects touch each other and backgrounds vary across scenes, our approach can still separate objects successfully. Furthermore, the generated images are shape and and reflect scene structures faithfully even on the very challenging Arrow Room dataset with subtle scene structures.

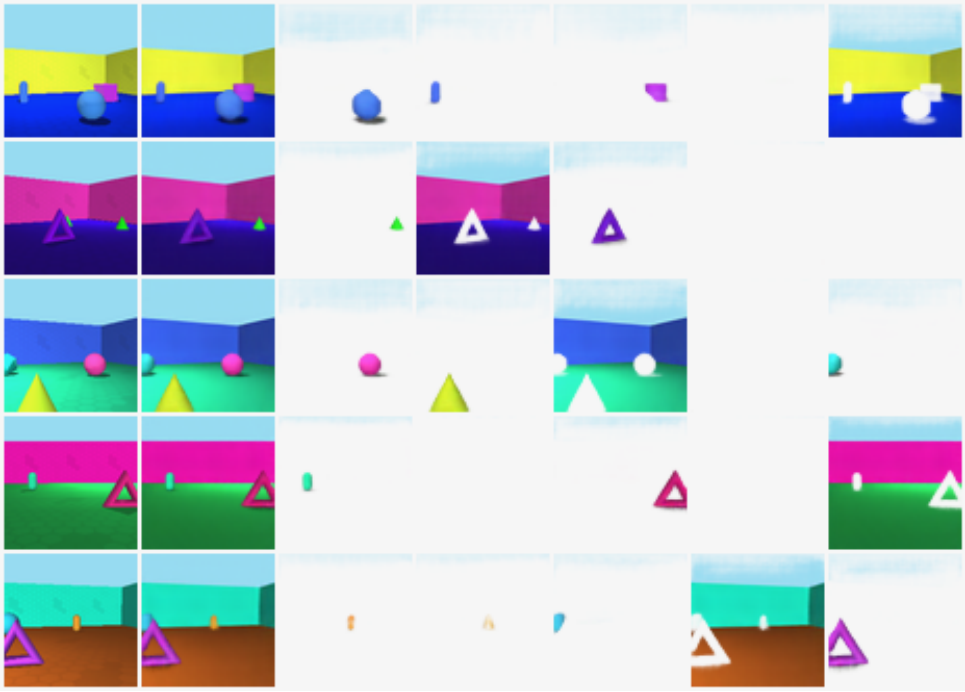

Figure 3.7: Additional decomposition resulst of Slot-VAE (ObjectsRoom dataset).

3.6.2. ADDITIONAL EXPERIMENTS

We conduct additional experiments to compare our proposed model Slot-VAE with vanilla slot attention in terms of decomposition performance (ARI score) on the ObjectsRoom, ShapeStacks datasets and the CLEVR6 dataset (the dataset that is used in the vanilla slot attention paper). As we can see in Table 3.4 that our proposed Slot-VAE achieves comparable decomposition performance and reconstruction performance (at least without significant degradation) to slot attention, while, standing out for being able to generate novel scenes.

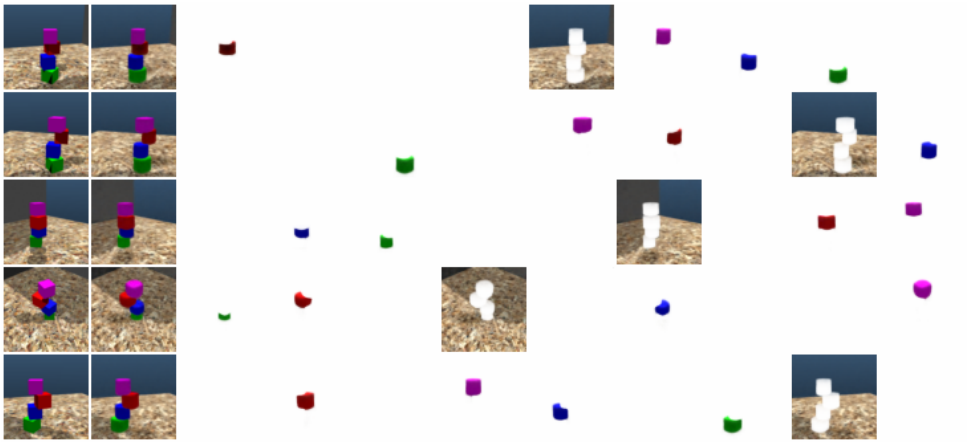

Figure 3.8: Additional decomposition resulst of Slot-VAE (ShapeStacks dataset).

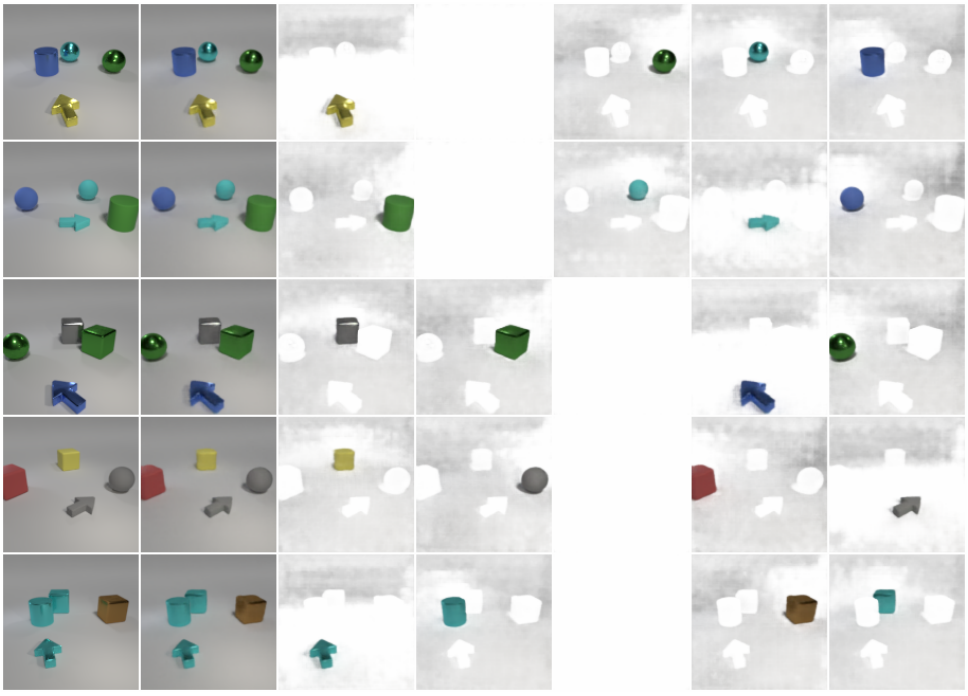

Figure 3.9: Additional decomposition resulst of Slot-VAE (ShapeStacks dataset).

3.6.3. MODEL DETAILS

In this section, we introduce the implementation details of Slot-VAE. As shown in Fig. 3.1, Slot-VAE has two parallel paths to train a two-layer hierarchical VAE model, which

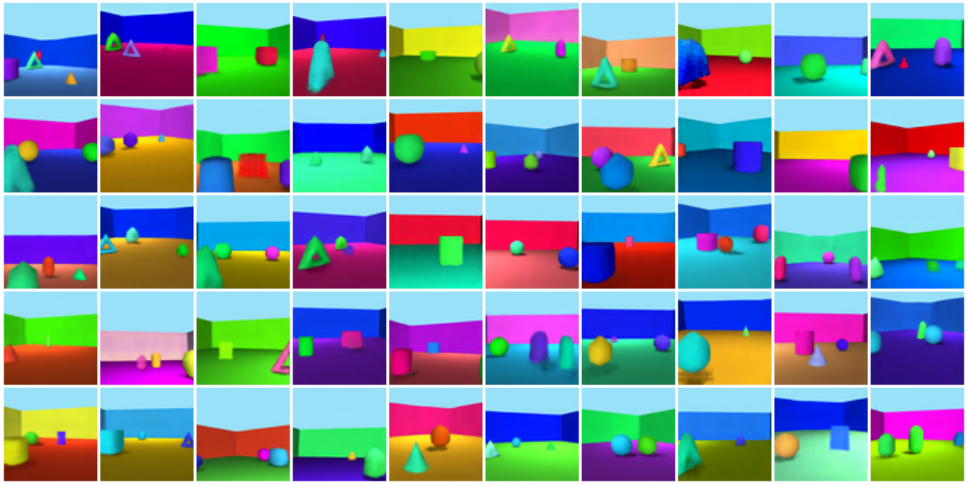

Figure 3.10: Additional generation resulst of Slot-VAE (Arrow Room dataset).

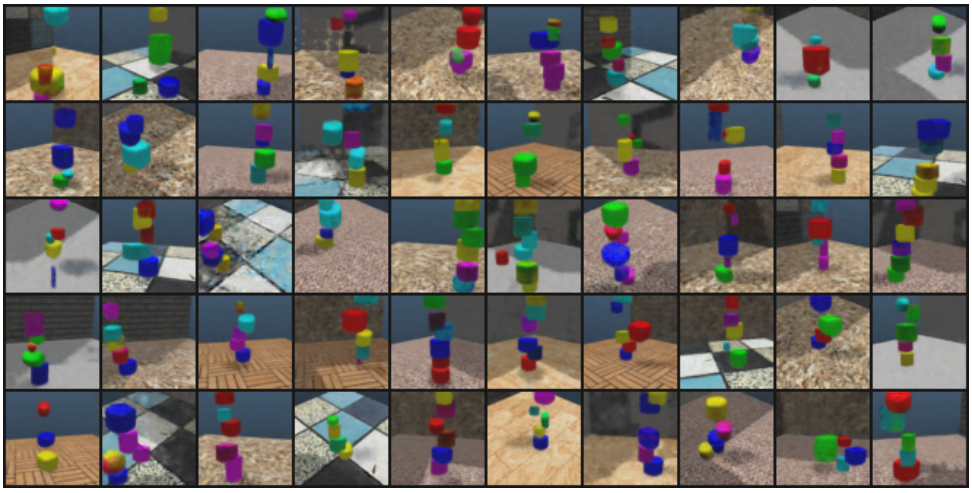

Figure 3.11: Additional generation resulst of Slot-VAE (ShapeStacks dataset).

Table 3.4: ARI (↑) score of Slot-VAE and Slot Attention.

	ObjectsRoom	ShapeStacks	CLEVR6
Slot-VAE	0.79±0.01	0.80±0.01	0.97±0.01
Slot Attention	0.81±0.02	0.80±0.00	0.98±0.01

Figure 3.12: Additional generation resulst of Slot-VAE (Arrow Room dataset).

mainly includes the following four modules.

CNN backbone. Before inferring the global latent representation and slot representations, the input image is first fed into a convolutional neural network to extract relatively high-level features. This convolutional neural network has 4 layers, each layer is with kernel size 5 and stride 1 and the final layer has 64 channels. The obtained feature map $\mathbf{f}_x$ still has image-sized dimensions and each feature (channel) has a dimension of 64, i.e., the dimension is $H \times W \times 64$. Soft position embedding are then added to the feature map to provide position information for the following modules.

Slot Attention Module. On the first path, we adopt the slot attention module [28] for object-centric representation learning. We include the details for self-containing purpuse. To prepare for slot learning, the feature map $\mathbf{f}_x$ is first flattened into vectors $\mathbf{f}_{input}$ with dimension $(H \times W) \times 64$. To cluster the feature vectors into object components, the clustering center, i.e., slots, should be initialized first. The initialization values for object slots are from Gaussian distribution respectively, i.e., $\mathbf{s}_{1:K} \sim \mathcal{N}(\mu, \mathrm{diag}(\sigma)) \in \mathbb{R}^{K \times 64}$, where μ and σ are learnable parameters. These slots are then updated iteratively to compete for explaining feature vectors $\mathbf{f}_{input}$. The slot competition is achieved via a softmax-based attention mechanism :$\mathrm{attn}_{i,j} := \frac{\exp(M_{i,j})}{\sum_l \exp(M_{i,l})}$, where $M := \frac{1}{\sqrt{D}} k(\mathbf{f}_{input}) \cdot q(\mathbf{s}_{1:K})^T \in \mathbb{R}^{(H \times W) \times K}$, and k and q are learnable linear mappings $\mathbb{R}^{D \to D}$ as commonly used in the attention mechanism, and $\sqrt{D}$ is a fixed value for softmax temperature. With the calculated attention scores $\mathrm{attn}_{i,j}$, image feature vectors $\mathbf{f}_{input}$ are aggregated via weighted mean: $\mathrm{updates} := \mathbf{W}^T \cdot v(\mathbf{f}_{input}) \in \mathbb{R}^{K \times D}$, where $\mathbf{W}_{i,j} := \mathrm{attn}_{i,j} / (\sum_{l=1}^{N} \mathrm{attn}_{l,j})$, and v is also learnable linear mappings similar to k and q. The update of slots in each iteration is completed via a learnable mapping parameterized by a Gated Recurrent Unit (GRU): $\mathbf{s}_{1:K} \leftarrow \mathrm{GRU}(\mathbf{s}_{1:K}, \mathrm{updates})$. The attention computation and updating are repeated 3

iterations to output final object-centric representations $\mathbf{s}_{1:K}$. Finally we obtain K vectors $\mathbf{s}_k$ each of dimension 64. To infer probabilistic random variables from $\mathbf{s}_k$, a MLP is used to map $\mathbf{s}_k$ to $\mathbf{z}^s_k$. This MLP is implemented with two layers with the first layer followed by a RELU layer. To be emphasized, the MLP is shared across $\mathbf{s}_k$, to encourage common formats of objet representations. The obtained object-centric latent vector $\mathbf{z}^s_k$ is still with a dimension of 64.

Global Auto-Encoding Module. To learn a global latent vector, the CNN backbone outputs $\mathbf{f}_x$ needs to be encoded by an encoder. Depending on the chosen prior distribution of the global latent vector, the encoder could have different structures. In the case that the global prior is Normal distribution, the encoder can be common ones used in vanilla VAE. Specifically, the $(H \times W) \times 64$ feature map is further flattened into one dimension, i.e., $(H \times W \times 64) \times 1$. Then a three-layer MLP, severing as an information bottleneck, reduces the dimension of obtained feature map to $\mathbf{z}^g$ of dimension 32×1. The obtained $\mathbf{z}^g$ can be decoded with deconvolutional neural nets back to the dimension of $(H \times W) \times 64$, trying to reconstruct the feature map. However, since the decoded feature map $\mathbf{f}$ is not used to recover image, rather generated object-centric latent vectors $\mathbf{z}^s_k$, there is no guarantee that $\mathbf{f}$ will be the same as $\mathbf{f}_x$. But with proper training, they should be close to each other. In summary, the auto-encoding structure is the same as commonly used VAE architecture. Another case for this global auto-encoding module is that a more powerful Strucdraw prior is used for the global latent vector learning. In that case, $\mathbf{z}^g$ is inferred autoregressively, the detail of such an encoder architecture could be found in [22]. Along the path of global auto-encoding, the obtained $\mathbf{z}^g$ of dimension 32 is then fed into a slot attention module. This slot attention module has exactly the same architecture as the one on the first path. The two slot attention modules share parameters.

Object Component Decoder. We choose the SBD decoder [57] as part of the object component decoder in our model. Different from [28] and [27] where a pure SBD is used, we combine SBD decoder with deconvolutional neural networks to balance the capacity of the decoder. Specifically, each object-centric latent vectors $\mathbf{z}^s_k$ of dimension 64 is first broadcast to a feature with shape $8 \times 8 \times 64$. Then this feature is decoded with deconvolutional neural nets with each layer having stride 2 and kernel size 5, to reconstruct an image-sized tensor with an additional channel as the mixing masks. The final output of the decoder has the shape $H \times W \times 4$. This decoder is shared across object-centric latent vectors $\mathbf{z}^s_k$.

3.6.4. EXPERIMENT DETAILS

Learning setups. Learning rate warm-up is important for object-centric representation learning as acknowledged by prior works. In the experiments, 10000 warm-up steps are used. For *ObjectRoom*, the batch size is 64, and the learning rate is 0.0004; for *ShapeStacks*, the batch size is 32, and the learning rate is 0.0001; and for *Arrow Room*, the batch size is 32, and the learning rate is 0.0001 in the early training steps and is decreased to 0.00005 after object-centric representations show up for stable training purpose.

Hyperparameter for the KL term $D_{\mathbf{KL}}\big[q_\phi(\mathbf{z}^g \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z}^g)\big]$. During training, we empirically find that multiplying $D_{\mathrm{KL}}\big[q_\phi(\mathbf{z}^g \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z}^g)\big]$ with a small hyperparameter

β helps $\mathbf{z}^g$ to encode meaningful scene representations. When β is too large, $\mathbf{z}^g$ tends to totally collapse to $p_\theta(\mathbf{z}^g)$, i.e., normal distribution. In the experiments, for *ObjectRoom*, β is 0.01; for *ShapeStacks*, β is 0.1; and for *Arrow Room*, β is 0.1.

REFERENCES

[1] Y. Wang, L. Liu, and J. Dauwels. “Slot-vae: Object-centric scene generation with slot attention”. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 36020–36035.

[2] A. Yuille and D. Kersten. “Vision as Bayesian inference: analysis by synthesis?” In: *Trends in cognitive sciences* 10.7 (2006), pp. 301–308.

[3] S. M. Frankland and J. D. Greene. “Concepts and compositionality: in search of the brain’s language of thought”. In: *Annual review of psychology* 71 (2020), pp. 273–303.

[4] P. N. Johnson-Laird. *Mental models: Towards a cognitive science of language, inference, and consciousness*. 6. Harvard University Press, 1983.

[5] D. Ha and J. Schmidhuber. “World models”. In: *arXiv preprint arXiv:1803.10122* (2018).

[6] B. Wu, S. Nair, R. Martin-Martin, L. Fei-Fei, and C. Finn. “Greedy hierarchical variational autoencoders for large-scale video prediction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 2318–2328.

[7] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio. “Toward causal representation learning”. In: *Proceedings of the IEEE* 109.5 (2021), pp. 612–634.

[8] P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum. “Simulation as an engine of physical scene understanding”. In: *Proceedings of the National Academy of Sciences* 110.45 (2013), pp. 18327–18332.

[9] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman. “Building machines that learn and think like people”. In: *Behavioral and brain sciences* 40 (2017), e253.

[10] A. Geiger, P. Lenz, and R. Urtasun. “Are we ready for autonomous driving? the kitti vision benchmark suite”. In: *2012 IEEE conference on computer vision and pattern recognition*. IEEE. 2012, pp. 3354–3361.

[11] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. “The cityscapes dataset for semantic urban scene understanding”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 3213–3223.

[12] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. “A simple neural network module for relational reasoning”. In: *Advances in neural information processing systems* 30 (2017).

[13] C. Devin, P. Abbeel, T. Darrell, and S. Levine. “Deep object-centric representations for generalizable robot learning”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2018, pp. 7111–7118.

[14] K. Greff, S. Van Steenkiste, and J. Schmidhuber. “On the binding problem in artificial neural networks”. In: *arXiv preprint arXiv:2012.05208* (2020).

[15] D. Mambelli, F. Träuble, S. Bauer, B. Schölkopf, and F. Locatello. “Compositional multi-object reinforcement learning with linear relation networks”. In: *arXiv preprint arXiv:2201.13388* (2022).

[16] D. P. Kingma and M. Welling. “Auto-encoding variational bayes”. In: *arXiv preprint arXiv:1312.6114* (2013).

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

[18] S. Eslami, N. Heess, T. Weber, Y. Tassa, D. Szepesvari, G. E. Hinton, *et al.* “Attend, infer, repeat: Fast scene understanding with generative models”. In: *Advances in neural information processing systems* 29 (2016).

[19] E. Crawford and J. Pineau. “Spatiial Invariant Unsupervised Object Detection with Convolutional Neural Networks”. In: *Thirty-Third AAAI Conference on Artificial Intelligence*. 2019.

[20] Z. Lin, Y.-F. Wu, S. V. Peri, W. Sun, G. Singh, F. Deng, J. Jiang, and S. Ahn. “Space: Unsupervised object-oriented scene representation via spatial attention and decomposition”. In: *arXiv preprint arXiv:2001.02407* (2020).

[21] J. Jiang, S. Janghorbani, G. De Melo, and S. Ahn. “Scalor: Generative world models with scalable object representations”. In: *arXiv preprint arXiv:1910.02384* (2019).

[22] J. Jiang and S. Ahn. “Generative neurosymbolic machines”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 12572–12582.

[23] M. Engelcke, O. Parker Jones, and I. Posner. “Genesis-v2: Inferring unordered object representations without iterative refinement”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 8085–8094.

[24] P. Emami, P. He, S. Ranka, and A. Rangarajan. “Slot Order Matters for Compositional Scene Understanding”. In: *arXiv preprint arXiv:2206.01370* (2022).

[25] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner. “Monet: Unsupervised scene decomposition and representation”. In: *arXiv:1901.11390* (2019).

[26] K. Greff, R. L. Kaufmann, R. Kabra, N. Watters, C. Burgess, D. Zoran, L. Matthey, M. Botvinick, and A. Lerchner. “Multi-object representation learning with iterative variational inference”. In: *arXiv preprint arXiv:1903.00450* (2019).

[27] M. Engelcke, A. R. Kosiorek, O. P. Jones, and I. Posner. “Genesis: Generative scene inference and sampling with object-centric latent representations”. In: *arXiv preprint arXiv:1907.13052* (2019).

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

[29] K. Greff, S. Van Steenkiste, and J. Schmidhuber. "Neural expectation maximization". In: *Advances in Neural Information Processing Systems* 30 (2017).

[30] A. Kosiorek, H. Kim, Y. W. Teh, and I. Posner. "Sequential attend, infer, repeat: Generative modelling of moving objects". In: *Advances in Neural Information Processing Systems* 31 (2018).

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

[32] G. Singh, F. Deng, and S. Ahn. "Illiterate dall-e learns to compose". In: *International Conference on Learning Representations*. 2021.

[33] T. Kipf, G. F. Elsayed, A. Mahendran, A. Stone, S. Sabour, G. Heigold, R. Jonschkowski, A. Dosovitskiy, and K. Greff. "Conditional object-centric learning from video". In: *arXiv preprint arXiv:2111.12594* (2021).

[34] M. Seitzer, M. Horn, A. Zadaianchuk, D. Zietlow, T. Xiao, C.-J. Simon-Gabriel, T. He, Z. Zhang, B. Schölkopf, T. Brox, *et al.* "Bridging the gap to real-world object-centric learning". In: *arXiv preprint arXiv:2209.14860* (2022).

[35] G. Singh, Y. Kim, and S. Ahn. *Neural Systematic Binder*. 2022. DOI: 10.48550/ARXIV.2211.01177. URL: https://arxiv.org/abs/2211.01177.

[36] G. Elsayed, A. Mahendran, S. Van Steenkiste, K. Greff, M. C. Mozer, and T. Kipf. "Savi++: Towards end-to-end object-centric learning from real-world videos". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 28940–28954.

[37] M. Engelcke, O. P. Jones, and I. Posner. "Reconstruction bottlenecks in Object-Centric generative models". In: *arXiv preprint arXiv:2007.06245* (2020).

[38] F. Deng, Z. Zhi, D. Lee, and S. Ahn. "Generative scene graph networks". In: *International Conference on Learning Representations*. 2021.

[39] S. Van Steenkiste, K. Kurach, J. Schmidhuber, and S. Gelly. "Investigating object compositionality in generative adversarial networks". In: *Neural Networks* 130 (2020), pp. 309–325.

[40] T. H. Nguyen-Phuoc, C. Richardt, L. Mai, Y. Yang, and N. Mitra. "Blockgan: Learning 3d object-aware scene representations from unlabelled images". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6767–6778.

[41] Y. Liao, K. Schwarz, L. Mescheder, and A. Geiger. "Towards unsupervised learning of generative models for 3d controllable image synthesis". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 5871–5880.

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

[43] S. Ehrhardt, O. Groth, A. Monszpart, M. Engelcke, I. Posner, N. Mitra, and A. Vedaldi. “RELATE: Physically plausible multi-object scene synthesis using structured latent spaces”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 11202–11213.

[44] Y. Du and I. Mordatch. “Implicit Generation and Generalization in Energy-Based Models”. In: *CoRR* abs/1903.08689 (2019). arXiv: 1903.08689. URL: http://arxiv.org/abs/1903.08689.

[45] Y. Du, S. Li, and I. Mordatch. “Compositional Visual Generation with Energy Based Models”. In: *Advances in Neural Information Processing Systems*. 2020.

[46] Y. Du, S. Li, J. B. Tenenbaum, and i. Mordatch. “Improved Contrastive Divergence Training of Energy Based Models”. In: *Proceedings of the 38th international conference on Machine learning*. ACM. 2021.

[47] N. Liu, S. Li, Y. Du, J. Tenenbaum, and A. Torralba. “Learning to compose visual relations”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 23166–23178.

[48] N. Liu, S. Li, Y. Du, A. Torralba, and J. B. Tenenbaum. “Compositional Visual Generation with Composable Diffusion Models”. In: *arXiv preprint arXiv:2206.01714* (2022).

[49] D. J. Rezende and F. Viola. “Taming vaes”. In: *arXiv preprint arXiv:1810.00597* (2018).

[50] H. Fu, C. Li, X. Liu, J. Gao, A. Celikyilmaz, and L. Carin. “Cyclical annealing schedule: A simple approach to mitigating KL vanishing”. In: *NAACL HLT 2019-2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies-Proceedings of the Conference*. Association for Computational Linguistics (ACL). 2019, pp. 240–250.

[51] R. Kabra, C. Burgess, L. Matthey, R. L. Kaufman, K. Greff, M. Reynolds, and A. Lerchner. *Multi-Object Datasets*. https://github.com/deepmind/multi-object-datasets/. 2019.

[52] O. Groth, F. B. Fuchs, I. Posner, and A. Vedaldi. “Shapestacks: Learning vision-based physical intuition for generalised object stacking”. In: *Proceedings of the european conference on computer vision (eccv)*. 2018, pp. 702–717.

[53] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. “beta-vae: Learning basic visual concepts with a constrained variational framework”. In: *International conference on learning representations*. 2017.

[54] R. T. Q. Chen, X. Li, R. Grosse, and D. Duvenaud. “Isolating Sources of Disentanglement in Variational Autoencoders”. In: *Advances in Neural Information Processing Systems*. 2018.

[55] L. Hubert and P. Arabie. “Comparing partitions”. In: *Journal of classification* 2 (1985), pp. 193–218.

[56] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. “Gans trained by a two time-scale update rule converge to a local nash equilibrium”. In: *Advances in Neural Information Processing Systems*. 2017, pp. 6626–6637.

[57] N. Watters, L. Matthey, C. P. Burgess, and A. Lerchner. “Spatial broadcast decoder: A simple architecture for learning disentangled representations in vaes”. In: *arXiv preprint arXiv:1901.07017* (2019).

[58] J. Jiang, F. Deng, G. Singh, and S. Ahn. “Object-Centric Slot Diffusion”. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 8563–8601.

4

Compositional Scene Decomposition

In the previous chapter, we demonstrated how to build compositional generative models by using object-centric approaches that model images as a masked summation of multiple objects, and illustrated this enables strong in-distribution generalization in unconditional scene generation tasks. However, for many other scene generation tasks such as combining visual concepts from different images, more flexible object representations than masks may be required. This is because objects and backgrounds from different images can have mismatched masks, which can lead to unnatural image generation when they are stitched together. Furthermore, object-centric approaches typically decompose images into local factors, such as objects, while struggling to capture global factors, such as lighting, as well as their high-level relationships that collectively compose an image. In this chapter, we propose a method that is capable of decomposing an image into not only local factors but also global factors. Our proposed approach, Decomp Diffusion, is an unsupervised method which, when given a single image, infers a set of different components in the image, each represented by a diffusion model. We demonstrate how components can capture different factors of the scene, ranging from global scene descriptors like shadows or facial expression to local scene descriptors like constituent objects. We further illustrate how inferred factors can be flexibly composed, even with factors inferred from other models, to generate a variety of scenes sharply different than those seen in training time.

This chapter is based on the paper published in Proceedings of the 41st International Conference on Machine Learning, PMLR 235:46823-46842, (2024) [1].

4.1. INTRODUCTION

4.1.1. BACKGROUND AND MOTIVATION

Humans have the remarkable ability to quickly learn new concepts, such as learning to use a new tool after observing just a few demonstrations [2]. This skill relies on the ability to combine and reuse previously acquired concepts to accomplish a given task [3]. This is particularly evident in natural language, where a limited set of words can be infinitely combined under grammatical rules to express various ideas and opinions [4]. In this work, we propose a method to discover compositional concepts from images in an unsupervised manner, which may be flexibly combined both within and across different image modalities.

Prior works on unsupervised compositional concept discovery may be divided into two separate categories. One line of approach focuses on discovering a set of global, holistic factors by representing data points in fixed factorized vector space [5–8]. Individual factors, such as facial expression or hair color, are represented as independent dimensions of the vector space, with recombination between concepts corresponding to recombination between underlying dimensions. However, since the vector space has a fixed dimensionality, multiple instances of a single factor, such as multiple different sources of lighting, may not be easily combined. Furthermore, as the vector space has a fixed underlying structure, individual factored vector spaces from different models trained on different datasets may not be combined, e.g., the lighting direction in one dataset with the foreground of an image from another.

An alternative approach decomposes a scene into a set of different underlying "object" factors. Each individual factor represents a separate set of pixels in an image defined by a disjoint segmentation mask [9–12]. Composition between different factors then corresponds to composing their respective segmentation masks. However, this method struggles to model higher-level relationships between factors, as well as multiple global factors that collectively affect the same image.

Recently, COMET [13] proposes to instead decompose a scene into a set of factors represented as *energy functions*. Composition between factors corresponds to solving for a minimal energy image subject to each energy function. Each individual energy function can represent global concepts such as facial expression or hair color as well as local concepts such as objects. However, COMET is unstable to train due to second-order gradients, and often generates blurry images.

In this chapter, we leverage the close connection between Energy-Based Models [14, 15] and diffusion models [16, 17] and propose Decomp Diffusion, an approach to decompose a scene into a set of factors, each represented as separate diffusion models. Composition between factors is achieved by sampling images from a composed diffusion distribution [18, 19], as illustrated in Figure 5.1. Similar to composition between energy functions, this composition operation allows individual factors to represent both global and local concepts and further enables the recombination of concepts across models and datasets. However, unlike the underlying energy decomposition objective of COMET, Decomp Diffusion may directly be trained through denoising, a stable and less expensive learning objective, and leads to higher resolution images.

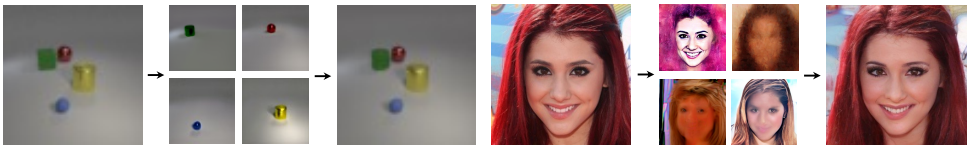

Figure 4.1: **Image Decomposition with Decomp Diffusion.** Our unsupervised method can decompose an input image into both local factors, such as objects (**Left**), and global factors (**Right**), such as facial features. Additionally, our approach can combine the deduced factors for image reconstruction.

4.1.2. CHAPTER CONTRIBUTIONS

We contribute the following in this chapter: First, we present Decomp Diffusion, an approach using diffusion models to decompose scenes into a set of different compositional concepts which substantially outperforms prior work using explicit energy functions. Second, we show that Decomp Diffusion is able to successfully decompose scenes into both global concepts as well as local concepts. Finally, we show that concepts discovered by Decomp Diffusion generalize well, and are amenable to compositions across different modalities of data, as well as components discovered by other instances of Decomp Diffusion.

4.2. RELATED WORK

Compositional Generation. Existing work on compositional generation study either modifying the underlying generative process to focus on a set of specifications [20–24], or composing a set of independent models specifying desired constraints [18, 19, 25–28]. Similar to [29], our work aims discover a set of compositional components from an unlabeled dataset of images which may further be integrated with compositional operations from [18, 19].

Unsupervised Decomposition. Unsupervised decomposition focuses on discovering a global latent space which best describes the input space [7, 8, 30–34]. In contrast, our approach aims to decompose data into multiple different compositional vector spaces, which allow us to both compose multiple instances of one factor together, as well as compose factors across different datasets. The most similar work in this direction is COMET [13], but unlike COMET we decompose images into a set of different diffusion models, and illustrate how this enables higher fidelity and more scalable image decomposition.

Unsupervised Object-Centric Learning. Object-centric learning approaches seek to decompose a scene into objects [9, 10, 12, 35–38], but unlike our method, they are unable to model global factors that collectively affect an image. Furthermore, although some approaches adopt a diffusion model for better local factor decomposition [39, 40], they only use the diffusion model as a decoder and still rely on a Slot Attention encoder for decomposition. In contrast, our approach is not limited by a specific encoder architecture because factor discovery is performed by modeling a composition

of energy landscapes through the connection between diffusion models and EBMs.

Diffusion-Based Concept Learning. Recent diffusion-based approaches often learn to acquire concepts by optimizing token embeddings with a collection of similar images [41–48], and so can be deemed supervised methods. The use of segmentation in decomposition has been explored in other methods, for example using through segmentation masks [49–51] or text captions [52], while our decomposition approach is completely unsupervised. The most relevant work to ours, [53] learns to decompose a set of images into a basis set of components using a pretrained text-to-image generative model in an unsupervised manner. However, our work aims to discover components per individual image.

4.3. UNSUPERVISED DECOMPOSITION OF IMAGES INTO ENERGY FUNCTIONS

In this section, we introduce background information about COMET [13], which our approach extends. COMET infers a set of latent factors from an input image, and uses each inferred latent to define a separate energy function over images. To generate an image that exhibits inferred concepts, COMET runs an optimization process over images on the sum of different energy functions.

In particular, given an image $\boldsymbol{x}_i \in \mathbb{R}^D$, COMET uses a learned encoder $\text{Enc}_\phi(\boldsymbol{x}_i)$ to infer a set of K different latents $\boldsymbol{z}_k \in \mathbb{R}^M$, where each latent $\boldsymbol{z}_k$ represents a different concept in an image. Both the image and latents are passed into an energy function $E_\theta(\boldsymbol{x}_i, \boldsymbol{z}_k) : \mathbb{R}^D \times \mathbb{R}^M \to \mathbb{R}$, which maps these variables to a scalar energy value.

Given a set of different factors $\boldsymbol{z}_k$, decoding these factors to an image corresponds to solving the optimization problem:

$$\arg\min_{\boldsymbol{x}} \sum_k E_\theta(\boldsymbol{x}; \boldsymbol{z}_k). \tag{4.1}$$

To solve this optimization problem, COMET runs an iterative gradient descent procedure from an image initialized from Gaussian noise. Factors inferred from either different images or even different models may likewise be decoded by optimizing the energy function corresponding to sum of energy function of each factor.

COMET is trained so that the K different inferred factors $\boldsymbol{z}_k$ from an input image $\boldsymbol{x}_i$ define K energy functions, so that the minimal energy state corresponds to the original image $\boldsymbol{x}_i$:

$$\mathcal{L}_{\text{MSE}}(\theta) = \left\| \arg\min_{\boldsymbol{x}} \left(\sum_k E_\theta(\boldsymbol{x}; \boldsymbol{z}_k) \right) - \boldsymbol{x}_i \right\|^2, \tag{4.2}$$

where $\boldsymbol{z}_k = \text{Enc}_\phi(\boldsymbol{x}_i)[k]$. The argmin of the sum of the energy functions is approximated by N steps of gradient descent

$$\boldsymbol{x}_i^N = \boldsymbol{x}_i^{N-1} - \gamma \nabla_{\boldsymbol{x}} \sum_k E_\theta(\boldsymbol{x}_i^{N-1}; \text{Enc}_\phi(\boldsymbol{x}_i)[k]), \tag{4.3}$$

where γ is the step size. Optimizing the training objective in Equation 4.2 corresponds to back-propagating through this optimization objective. The resulting

process is computationally expensive and unstable to train, as it requires computing second-order gradients.

4.4. COMPOSITIONAL IMAGE DECOMPOSITION WITH DIFFUSION MODELS

Next, we discuss how to decompose images into a set of composable diffusion models. We first discuss how diffusion models may be seen as parameterizing energy functions in Section 4.4.1. Then in Section 4.4.2, we describe how we use this connection in Decomp Diffusion to decompose images into a set of composable diffusion models.

4.4.1. DENOISING NETWORKS AS ENERGY FUNCTIONS

Denoising Diffusion Probabilistic Models (DDPMs) [16, 17] are a class of generative models that facilitate generation of images $\boldsymbol{x}_0$ by iteratively denoising an image initialized from Gaussian noise. Given a randomly sampled noise value $\epsilon \sim \mathcal{N}(0,1)$, as well as a set of t different noise levels $\epsilon^t = \sqrt{\beta_t}\epsilon$ added to a clean image $\boldsymbol{x}_i$, a denoising model ϵ_θ is trained to denoise the image at each noise level t:

$$\mathcal{L}_{\text{MSE}} = \|\epsilon - \epsilon_\theta(\sqrt{1-\beta_t}\boldsymbol{x}_i + \sqrt{\beta_t}\epsilon, t)\|_2^2. \tag{4.4}$$

In particular, the denoising model learns to estimate a gradient field of natural images, describing the direction that noisy images $\boldsymbol{x}^t$ with noise level t should be refined toward to become natural images [17]. As discussed in both [18, 19], this gradient field also corresponds to the gradient field of an energy function

$$\epsilon_\theta(\boldsymbol{x}^t, t) = \nabla_{\boldsymbol{x}} E_\theta(\boldsymbol{x}) \tag{4.5}$$

that represents the relative log-likelihood of a datapoint.

To generate an image from the diffusion model, a sample $\boldsymbol{x}^T$ at noise level T is initialized from Gaussian noise $\mathcal{N}(0,1)$ and then iteratively denoised through

$$\boldsymbol{x}^{t-1} = \boldsymbol{x}^t - \gamma\epsilon_\theta(\boldsymbol{x}^t, t) + \xi, \quad \xi \sim \mathcal{N}\left(0, \sigma_t^2 I\right), \tag{4.6}$$

where σ_t^2 is the variance according to a variance schedule and γ is the step size[1]. As we introduced in Chapter 2.3.3, this directly corresponds to the noisy energy optimization procedure

$$\boldsymbol{x}^{t-1} = \boldsymbol{x}^t - \gamma\nabla_{\boldsymbol{x}} E_\theta(\boldsymbol{x}^t) + \xi, \quad \xi \sim \mathcal{N}\left(0, \sigma_t^2 I\right). \tag{4.7}$$

The functional form of Equation 4.7 is very similar to Equation 4.3, and illustrates how sampling from a diffusion model is similar to optimizing a learned energy function $E_\theta(\boldsymbol{x})$ that parameterizes the relative negative log-likelihood of the data density.

When we train a diffusion model to recover a conditional data density that consists of a single image $\boldsymbol{x}_i$, i.e., when we are autoencoding an image given an inferred

[1] An linear decay $\frac{1}{\sqrt{1-\beta_t}}$ is often also applied to the output $\boldsymbol{x}^{t-1}$ for sampling stability.

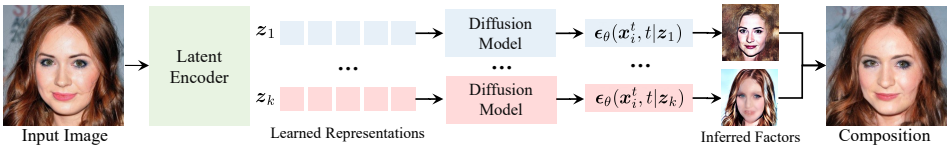

Figure 4.2: **Compositional Image Decomposition**. We learn to decompose each input image into a set of denoising functions $\{\epsilon_\theta(x_i^t, t, |z_k)\}$ representing K factors, which can be composed to reconstruct the input.

intermediate latent z, then the denoising network directly learns an $\epsilon_\theta(x, t, z)$ that estimates gradients of an energy function $\nabla_x E_\theta(x, z)$. This energy function has minimum

$$x_i = \arg\min_x E_\theta(x, z), \tag{4.8}$$

as the highest log-likelihood datapoint will be x_i. The above equivalence suggests that we may directly use diffusion models to parameterize the unsupervised decomposition of images into the energy functions discussed in Section 4.3.

4.4.2. DECOMPOSITIONAL DIFFUSION MODELS

In COMET, given an input image x_i, we are interested in inferring a set of different latent energy functions $E_\theta(x, z_k)$ such that

$$x_i = \arg\min_x \sum_k E_\theta(x, z_k).$$

Using the equivalence between denoising networks and energy function discussed in Section 4.4.1 to recover the desired set of energy functions, we may simply learn a set of different denoising functions to recover an image x_i using the objective:

$$\mathcal{L}_{\text{MSE}} = \left\| \epsilon - \sum_k \epsilon_\theta \left(\sqrt{1-\beta_t} x_i + \sqrt{\beta_t}\epsilon, t, z_k \right) \right\|_2^2, \tag{4.9}$$

where each individual latent z_k is inferred by a jointly learned neural network encoder $\text{Enc}_\phi(x_i)[k]$. We leverage information bottleneck to encourage components to discover independent portions of x_i by constraining latent representations $z = \{z_1, z_2, \cdots, z_K\}$ to be low-dimensional. This resulting objective is simpler to train than that of COMET, as it requires only a single step denoising supervision and does not need computation of second-order gradients.

Reconstruction Training. As discussed in [17], the denoising network ϵ_θ may either be trained to directly estimate the starting noise ϵ or the original image x_i. These two predictions are functionally identical, as ϵ can be directly obtained by taking a linear combination of noisy image x^t and x_i. While standard diffusion training directly predicts ϵ, we find that predicting x_i and then regressing ϵ leads to better performance, as this training objective is more similar to autoencoder training.

Algorithm 1 Training Algorithm

Input: Encoder Enc_ϕ, denoising model ϵ_θ, components K, data distribution p_D
while not converged **do**
  $\boldsymbol{x}_i \sim p_D$
  $\triangleright$ *Extract components* $\boldsymbol{z}_k$ *from* $\boldsymbol{x}_i$
  $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K \leftarrow \text{Enc}_\phi(\boldsymbol{x}_i)$
  $\triangleright$ *Compute denoising direction*
  $\epsilon \sim \mathcal{N}(0,1), t \sim \text{Unif}(\{1, \ldots, T\})$
  $\boldsymbol{x}_i^t = \sqrt{1-\beta_t}\boldsymbol{x}_i + \sqrt{\beta_t}\epsilon$
  $\epsilon_{\text{pred}} \leftarrow \sum_k \epsilon_\theta(\boldsymbol{x}_i^t, t, \boldsymbol{z}_k)$
  $\triangleright$ *Optimize objective* $\mathcal{L}_{MSE}$ *wrt* $\zeta = \{\phi, \theta\}$*:*
  $\Delta\zeta \leftarrow \nabla_\zeta \|\epsilon_{\text{pred}} - \epsilon\|^2$
  Update ζ based on $\Delta\zeta$
end while

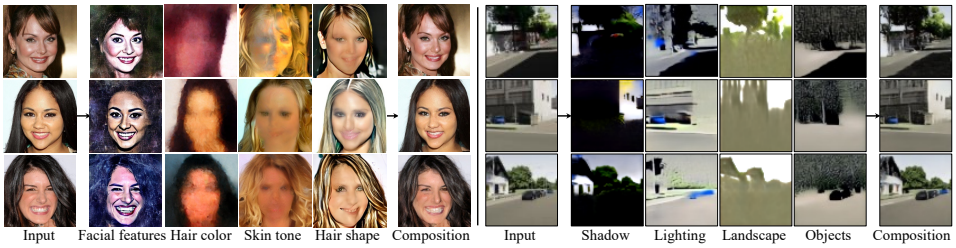

Figure 4.3: **Global Factor Decomposition.** Our method can enable global factor decomposition and reconstruction on CelebA-HQ (**Left**) and Virtual KITTI 2 (**Right**). *Note that discovered factors are labeled with posited factors.*

Once we have recovered these denoising functions, we may directly use the noisy optimization objective in Equation 4.7 to sample from compositions of different factors. The full training and sampling algorithm for our approach are shown in Algorithm 3 and Algorithm 2 respectively.

4.5. EXPERIMENTS

In this section, we evaluate the ability of our approach to decompose images. First, we assess decomposition of images into global factors of variation in Section 4.5.2. We next evaluate decomposition of images into local factors of variation in Section 4.5.3. We further investigate the ability of decomposed components to recombine across separate trained models in Section 4.5.4. Finally, we illustrate how our approach can be adapted to pretrained models in Section 4.5.5. We use datasets with a degree of consistency among the images, for example aligned face images, to ensure that they have common elements our approach could extract.

Algorithm 2 Image Generation Algorithm

Input: Diffusion steps T, denoising model ϵ_θ, latent vectors $\{z_1, \ldots, z_K\}$, step size γ
$x^T \sim \mathcal{N}(0, 1)$
for $t = T, \ldots, 1$ **do**
  ▷ *Sample Gaussian noise*
  $\xi \sim \mathcal{N}(0, 1)$
  ▷ *Compute denoising direction*
  $\epsilon_{\text{pred}} \leftarrow \sum_k \epsilon_\theta(x^t, t, z_k)$
  ▷ *Run noisy gradient descent*
  $x^{t-1} = \frac{1}{\sqrt{1-\beta_t}}(x^t - \gamma\epsilon_{\text{pred}} + \sqrt{\beta_t}\xi)$
end for

Figure 4.4: **Reconstruction comparison.** Our method can reconstruct input images with a high fidelity on CelebA-HQ.

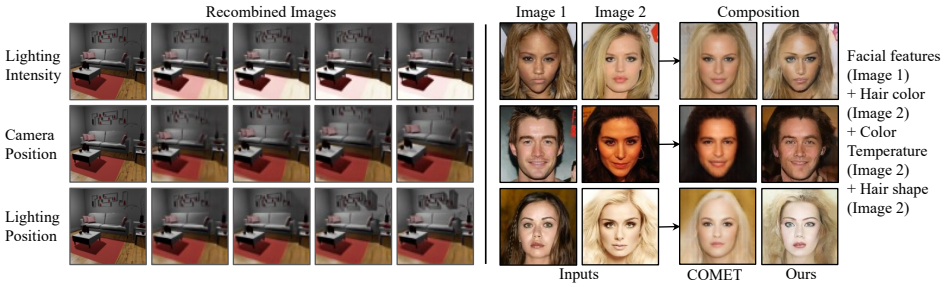

Figure 4.5: **Global Factor Recombination.** Recombination of inferred factors on Falcor3D and CelebA-HQ datasets. In Falcor3D (**Left**), we show image variations by varying inferred factors such as lighting intensity. In CelebA-HQ (**Right**), we recombine factors from two different inputs to generate novel face combinations.

4.5.1. QUANTITATIVE METRICS

For quantitative evaluation of image quality, we employ Fréchet Inception Distance (FID) [54], Kernel Inception Distance (KID) [55], and LPIPS [56] on images reconstructed from CelebA-HQ [57], Falcor3D [58], Virtual KITTI 2 [59], and CLEVR [60]. To evaluate disentanglement, we compute MIG [61] and MCC [62] on learned latent representation images on the Falcor3D dataset.

4.5.2. GLOBAL FACTOR DECOMPOSITION AND RECOMBINATION

Given a set of input images, we illustrate how our unsupervised approach can capture a set of global scene descriptors such as lighting and background and recombine them to construct image variations. We evaluate results in terms of image quality and disentanglement of global components.

Decomposition and Reconstruction. On the left-hand side of Figure 4.3, we show

Model	CelebA-HQ			Falcor3D			Virtual KITTI 2			CLEVR		
	FID ↓	KID ↓	LPIPS ↓	FID ↓	KID ↓	LPIPS ↓	FID ↓	KID ↓	LPIPS ↓	FID ↓	KID ↓	LPIPS ↓
β-VAE ($\beta = 4$)	107.29	0.107	0.239	116.96	0.124	0.075	196.68	0.181	0.479	316.64	0.383	0.651
MONet	35.27	0.030	0.098	69.49	0.067	0.082	67.92	0.043	0.154	60.74	0.063	0.118
COMET	62.64	0.056	0.134	46.37	0.040	0.032	124.57	0.091	0.342	103.84	0.119	0.141
Slot Attention	56.41	0.050	0.154	65.21	0.061	0.079	153.91	0.113	0.207	27.08	0.026	0.031
Hessian Penalty	34.90	0.021	–	322.45	0.479	–	116.91	0.084	–	25.40	0.016	–
GENESIS-V2	41.64	0.035	0.132	130.56	0.130	0.097	134.31	0.105	0.202	318.46	0.403	0.631
Ours	**16.48**	**0.013**	**0.089**	**14.18**	**0.008**	**0.028**	**21.59**	**0.008**	**0.058**	**11.49**	**0.011**	**0.012**

Table 4.1: **Image Reconstruction Evaluation.** We evaluate the quality of 64×64 reconstructed images using FID, KID and LPIPS on 10,000 images from 4 different datasets. Our method achieves the best performance.

how our approach decomposes CelebA-HQ face images into a set of factors. These factors can be qualitatively described as facial features, hair color, skin tone, and hair shape. To better visualize each factor's individual effect, we provide experiments in Figure 4.22 where factors are added one at a time to incrementally reconstruct the input image. In addition, we compare our method's performance on image reconstruction against existing baselines in Figure 4.4. Our method generates better reconstructions than COMET as well as other recent baselines, in that images are sharper and more similar to the input.

On the right side of Figure 4.3, we show how Decomp Diffusion infers factors such as shadow, lighting, landscape, and objects on Virtual KITTI 2. We can further compose these factors to reconstruct the input images, as illustrated in the rightmost column. Comparative decompositions from other methods can be found in Figure 4.19.

We also provide qualitative results to illustrate the effect of number of concepts K on CelebA-HQ and Falcor3D in Figure 4.17 and Figure 4.18, respecticely. As expected, using different K can lead to different sets of decomposed concepts being produced, but certain concepts are learned across different K, such as the facial features concepts in Figure 4.18.

Recombination. In Figure 4.5, we explore how factors can be flexibly composed by recombining decomposed factors from Falcor3D as well as from CelebA-HQ. On the left-hand side, we demonstrate how recombination can be performed on a source image by varying a target factor, such as lighting intensity, while preserving the other factors. This enables us to generate image variations using inferred factors such as lighting intensity, camera position, and lighting position.

On the right-hand side of Figure 4.5, we show how factors extracted from different faces can be recombined to generate a novel human face that exhibits the given factors. For instance, we can combine the facial features from one person with the hair shape of another to create a new face that exhibits the chosen properties. These results illustrate that our method can effectively disentangle images into global factors that can be recombined for novel generalization.

Quantitative results. To quantitatively compare different methods, we evaluate the visual quality of reconstructed images using the decomposed scene factors, as presented in Table 4.1. We observe that our method outperforms existing methods in terms of FID, KID, and LPIPS across datasets, indicating superior image reconstruction

Model	Dim (D)	β	Decoder Dist.	MIG ↑	MCC ↑
InfoGAN	64	–	–	2.48 ± 1.11	52.67 ± 1.91
β-VAE	64	4	Bernoulli	8.96 ± 3.53	61.57 ± 4.09
β-VAE	64	16	Gaussian	9.33 ± 3.72	57.28 ± 2.37
β-VAE	64	4	Gaussian	10.90 ± 3.80	66.08 ± 2.00
GENESIS-V2*	128	–	–	5.23 ± 0.02	63.83 ± 0.22
MONet	64	–	–	13.94 ± 2.09	65.72 ± 0.89
COMET	64	–	–	19.63 ± 2.49	76.55 ± 1.35
Ours	32	–	–	11.72 ± 0.05	57.67 ± 0.09
Ours	64	–	–	$\mathbf{26.45 \pm 0.16}$	$\mathbf{80.42 \pm 0.08}$
Ours	128	–	–	12.97 ± 0.02	80.27 ± 0.17
Ours*	128	–	–	16.57 ± 0.02	71.19 ± 0.15

Table 4.2: **Disentanglement Evaluation.** Mean and standard deviation of metrics across 3 random seeds on the Falcor3D dataset. enables better disentanglement according to 2 common disentanglement metrics. The asterisk (*) indicates that PCA is applied to project the output dimension to 64.

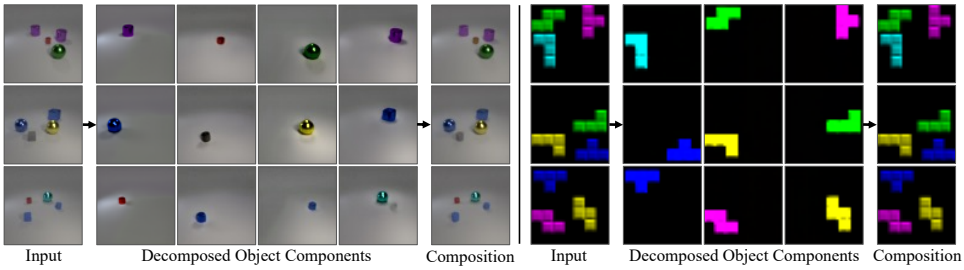

Figure 4.6: **Local Factor Decomposition.** Illustration of object-level decomposition on CLEVR (**left**) and Tetris (**right**). Our method can extract individual object components that can be reused for image reconstruction.

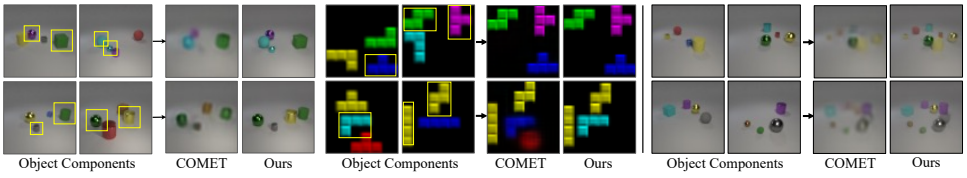

Figure 4.7: **Local Factor Recombination.** We recombine local factors from 2 images to generate composition of inferred object factors. On both CLEVR and Tetris (**Left**), we recombine inferred object components in the bounding box to generate novel object compositions. On CLEVR (**Right**), we compose all inferred factors to generalize up to 8 objects, though training images only contain 4 objetcs.

quality.

Finally, we evaluate the disentanglement of the given methods on the Falcor3D dataset. As shown in Table 4.2, Decomp Diffusion with dimension 64 achieves the best scores across disentanglement metrics, showing its effectiveness in capturing a

set of global scene descriptors. In addition, we evaluate our models with different latent dimensions of 32, 64, and 128 to investigate the impact of latent dimension. We find that our method achieves the best performance when using a dimension of 64. We posit that a smaller dimension may lack the capacity to encode all the information, thus leading to worse disentanglement. A larger dimension may be too large and fail to separate distinct factors. Thus, we apply PCA to project the output dimension 128 to 64 (last row), and we observe that it can boost the MIG performance but lower the MCC score.

Diffusion Parameterizations. We next analyze two choices of diffusion parameterizations for the model, predicting $\boldsymbol{x}_0$ or predicting the noise ϵ, in Table 4.3. We find that directly predicting the input $\boldsymbol{x}_0$ (3^{rd} and 6^{th} rows) outperforms the ϵ parametrization (1^{st} and 4^{th} row) on both CelebA-HQ and CLEVR datasets in terms of MSE and LPIPS [56]. This is due to using a reconstruction-based training procedure, as discussed in Section 4.4.2. We also compare using a single component to learn reconstruction (2^{nd} and 5^{th} rows) with our method (3^{rd} and 6^{th} rows), which uses multiple components for reconstruction. Our method achieves the best reconstruction quality as measured by MSE and LPIPS.

4.5.3. LOCAL FACTOR DECOMPOSITION AND RECOMBINATION

Given an input image with multiple objects, e.g., a purple cylinder and a green cube, we aim to factorize the input image into individual object components using object-level segmentation.

Decomposition and Reconstructions. We qualitatively evaluate local factor decomposition on object datasets such as CLEVR and Tetris in Figure 4.6. Given an image with multiple objects, our method can both isolate each individual object component as well as faithfully reconstruct the input image using the set of decomposed object factors. Note that since our method does not obtain an explicit segmentation mask per object, it is difficult to quantitatively assess segmentations (though empirically, we found our approach almost always correctly segments objects). We additionally provide results of factor-by-factor compositions, where images are generated by incrementally adding one component at a time, in Figure 4.23. These mirror the process of adding one object at a time to the scene and demonstrate that our method effectively learns local object-centric representations.

Recombination. To further validate our approach, we show how our method can recombine local factors from different input images to generate previously unseen image combinations. In Figure 4.7, we demonstrate how our method utilizes a subset of factors from each image for local factor recombination. On the left-hand side, we present novel object combinations generated by adding particular factorized energy functions from two inputs, shown within the bounding boxes, on both the CLEVR and Tetris datasets. On the right-hand side, we demonstrate how our method can recombine all existing local components from two CLEVR images into an unseen combination of 8 objects, even though each training image only consists of 4 objects. We illustrate that our approach is highly effective at recombining local factors to create novel image combinations.

Dataset	Multiple Components	Predict x_0	MSE ↓	LPIPS ↓	FID ↓	KID ↓
CelebA-HQ	Yes	No	105.003	0.603	155.46	0.141
	No	Yes	88.551	0.192	30.10	0.022
	Yes	Yes	**76.168**	**0.089**	**16.48**	**0.013**
CLEVR	Yes	No	56.179	0.3061	42.72	0.033
	No	Yes	26.094	0.2236	24.27	0.023
	Yes	Yes	**6.178**	**0.0122**	**11.54**	**0.010**

Table 4.3: **Ablations.** We analyze the impact of predicting x_0 or ϵ, as well as using multiple components or a single component. We compute pixel-wise MSE and LPIPS of reconstructions on both CLEVR and CelebA-HQ.

Figure 4.8: **Multi-modal Dataset Decomposition.** We show our method can capture a set of global factors that are shared between hybrid datasets such as KITTI and Virtual KITTI 2 scenes (**Left**), and CelebA-HQ and Anime faces (**Right**). Note that discovered factors are labeled with posited factors.

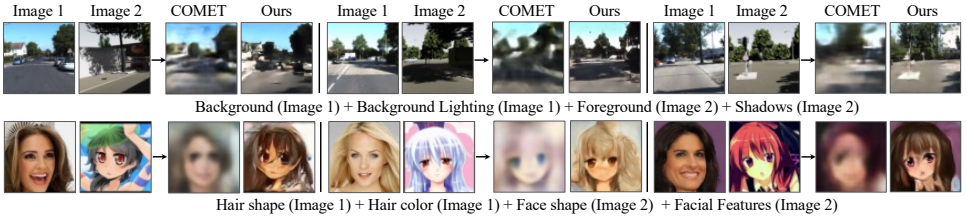

Figure 4.9: **Multi-modal Dataset Recombination.** Our method exhibits the ability to recombine inferred factors from various hybrid datasets. We can recombine different extracted factors to generate unique compositions of KITTI and Virtual KITTI 2 scenes (**Top**), and compositions of CelebA-HQ and Anime faces (**Bottom**).

4.5.4. IMAGE GENERATION WITH CROSS DATASET GENERALIZATION

We next assess the ability of our approach to extract and combine concepts across multiple datasets. We investigate the recombination of factors in multi-modal datasets, as well as the combination of separate factors from distinct models trained on different datasets.

Multi-modal Decomposition and Reconstruction. Multi-modal datasets, such as a dataset containing images from a photorealistic setting and an animated setting, pose

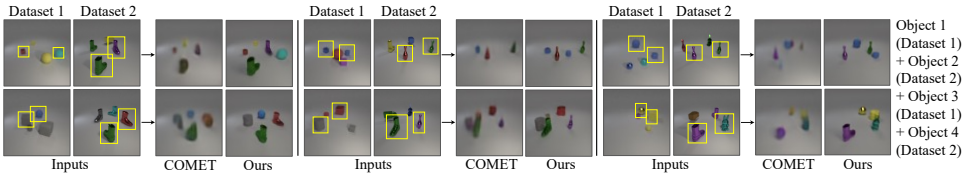

Figure 4.10: **Cross Dataset Recombination.** We further showcase our method's ability to recombine across datasets using 2 different models that train on CLEVR and CLEVR Toy, respectively. We compose inferred factors as shown in the bounding box from two different modalites to generate unseen compositions.

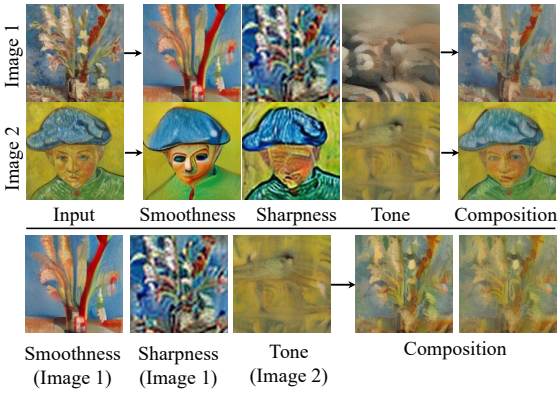

Figure 4.11: **Art Style Decomposition and Recombination.** Illustration of art style decomposition on a Van Gogh painting dataset. Our method can discover art components that capture different facets of the painting content. The discovered factors can be recombined across images to generate novel images.

a greater challenge for extracting common factors. Despite this, we demonstrate our method's success in this regard in Figure 4.8. The left-hand side exhibits the decomposition of images from a hybrid dataset comprising KITTI and Virtual KITTI into a set of global factors, such as background, lighting, and shadows. The right-hand side decomposes the two types of faces into a cohesive set of global factors including face shape, hair shape, hair color, and facial details, which can be utilized for reconstruction. This demonstrates our method's effectiveness in factorizing hybrid datasets into a set of factors.

Multi-modal Recombination. Furthermore, we assess the ability of our method to recombine obtained factors across multi-modal datasets, as illustrated in Figure 4.9. In the top half, in a hybrid KITTI and Virtual KITTI dataset, we recombine extracted factors from two distinct images to produce novel KITTI-like scenes, for instance incorporating a blue sky background with shadows in the foreground. In the bottom half, we demonstrate our method's ability to reuse and combine concepts to generate

unique anime faces, combining hair shapes and colors from a human face image with face shape and details from an anime face image.

Cross Dataset Recombination. Given one denoising model $\epsilon_1(x^t, t, z_k)$ trained on the CLEVR dataset and a second denoising model $\epsilon_2(x^t, t, z_n)$ trained on the CLEVR Toy dataset, we investigate combining local factors extracted from different modalities to generate novel combinations. To compose objects represented by z_1 and z_2 from one image in CLEVR dataset and objects represented by z_3 and z_4 from another image in the CLEVR Toy dataset, we sum the predicted individual noise corresponding to z_1, z_2, z_3, z_4, i.e., $\epsilon_{\text{pred}} = \epsilon_1(x^t, t, z_1) + \epsilon_1(x^t, t, z_2) + \epsilon_2(x^t, t, z_3) + \epsilon_2(x^t, t, z_4)$, and follow Algorithm 2 to generate a recombined image comprised of objects represented by z_1, z_2, z_3, z_4. In Figure 4.10, our method extracts object components in the bounding box from two images from different datasets, and then further combines them to generate unseen combinations of object components from different models. In Table 4.5, we provide the FID and KID scores of generated recombinations against the original CLEVR dataset and CLEVR Toy dataset. Our method outperforms COMET on both datasets, indicating the model can obtain better visual quality and more cohesive recombinations.

4

4.5.5. DECOMPOSITION AND RECOMBINATION WITH PRETRAINED MODELS

Finally, we illustrate that our approach can adopt pretrained diffusion models as a prior for visual decomposition to avoid training diffusion models from scratch. Specifically, we train the encoder Enc_ϕ and finetune Stable Diffusion model ϵ_θ together, in the same fashion as shown in Algorithm 3. The latent vectors inferred from the encoder are used as conditionings for the Stable Diffusion model to enable image decomposition and composition.

In our experiment, we train our model on a small dataset of 100 Van Gogh paintings for 1000 iterations. As shown in Figure 4.11, our method can decompose such images into a set of distinct factors, such as smoothness, sharpness, and color tone, which can be further recombined to generate unseen compositions like flowers with sharp edges and a yellow tone. Figure 4.11 also shows that our method can use weighted recombination to enhance or reduce individual factors. As an example, we give the tone factor two different weights in the recombination, which results in two images with different extents of yellow tone. This demonstrates that our method can be adapted to existing models efficiently.

4.6. SUMMARY

Conclusions. In this chapter, we present Decomp Diffusion and demonstrate its efficacy at decomposing images into both global factors of variation, such as facial expression, lighting, and background, and local factors, such as constituent objects. We further illustrate the ability of different inferred components to compose across multiple datasets and models. We also show that the proposed model can be adapted to existing pretrained models efficiently. We hope that our work inspires future research in unsupervised discovery of compositional representations in images.

Limitations. Our work has several limitations. First, our current approach decomposes images into a fixed number of factors that is specified by the user. While there are cases where the number of components is apparent, in many datasets the number is unclear or may be variable depending on the image. In Section 4.7.2, we study the sensitivity of our approach to the number of components. We find that we recover duplicate components when the number is too large, and subsets of components when it is too small. A principled approach to determine the ideal number of factors would be an interesting future line of work. In addition, factors discovered by our approach are not guaranteed be distinct from the original image or from each other, and if the latent encoder's embedding dimension is too large, each latent factor may capture the original image itself. Adding explicit regularization to enforce independence between latents would also be a potential area of future research.

Social Impact. Our proposed approach does not have immediate negative social impact in its current form since evaluation is carried out on standard datasets. However, our model's ability to generate facial features or objects in a zero-shot manner raises concerns about potential misuse for misinformation. Thus, advocating for responsible usage is crucial. Additionally, like many generative models, there is a risk of introducing biases related to gender or race depending on the training data. Therefore, careful attention must be paid to data collection and curation to mitigate such biases. Our approach can actually benefit many fields such as scene understanding, artwork generation, and robotics.

4.7. SUPPLEMENTARY

In this appendix, we present additional details on compositional scene decomposition. We illustrate additional qualitative results for various domains in Section 4.7.1 and additional experiments in Section 4.7.2. Next, we describe the model architecture for our approach in Section 4.7.3. Finally, we include experiment details on training datasets, baselines, training, and inference in Section 4.7.4.

4.7.1. ADDITIONAL QUALITATIVE RESULTS

We first provide additional results on global factor decomposition and recombination. We then give additional results on object-level decomposition and recombination. Finally, we provide more results that demonstrate cross-dataset generalization.

GLOBAL FACTOR DECOMPOSITION AND RECOMBINATION

Decomposition and Reconstruction. In Figure 4.12, we present supplemental image generations that demonstrate our approach's ability to capture global factors across different domains, such as human faces and scene environments. The left side of the figure displays how our method can decompose images into global factors like facial features, hair color, skin tone, and hair shape, which can be further composed to reconstruct the input images. On the right, we show additional decomposition and composition results using Virtual KITTI 2 images. Our method can effectively generate clear, meaningful global components from input images. In Figure 4.13, we show

decomposition and composition results on Falcor3D data. Through unsupervised learning, our approach can accurately discover a set of global factors that include foreground, background, objects, and lighting.

Recombination. Figure 4.14 showcases our approach's ability to generate novel image variations through recombination of inferred concepts. The left-hand side displays results of the recombination process on Falcor3D data, with variations on lighting intensity, camera position, and lighting position. On the right-hand side, we demonstrate how facial features and skin tone from one image can be combined with hair color and hair shape from another image to generate novel human face image combinations. Our method demonstrates great potential for generating diverse and meaningful image variations through concept recombination.

LOCAL FACTOR DECOMPOSITION AND RECOMBINATION

Decomposition and Reconstruction. We present additional results for local scene decomposition in Figure 4.15. Our proposed method successfully factorizes images into individual object components, as demonstrated in both CLEVR (**Left**) and Tetris (**Right**) object images. Our approach also enables the composition of all discovered object components for image reconstruction.

Recombination. We demonstrate the effectiveness of our approach for recombination of local scene descriptors extracted from multi-object images such as CLEVR and Tetris. As shown in Figure 4.16, our method is capable of generating novel combinations of object components by recombining the extracted components (shown within bounding boxes for easy visualization). Our approach can effectively generalize across images to produce unseen combinations.

CROSS DATASET GENERALIZATION

We investigate the recombination of factors inferred from multi-modal datasets, and the combination of separate factors extracted from distinct models trained on different datasets.

Multi-modal Decomposition and Reconstruction. We further demonstrate our method's capability to infer a set of factors from multi-modal datasets, , a dataset that consists of different types of images. On the left side of Figure 4.28, we provide additional results on a multi-modal dataset that consists of KITTI and Virtual KITTI 2. On the right side, we show more results on a multi-modal dataset that combines both CelebA-HQ and Anime datasets.

Multi-modal Recombination. In Figure 4.29, we provide additional recombination results on the two multi-modal datasets of KITTI and Virtual KITTI 2 on the left hand side of the Figure, and CelebA-HQ and Anime datasets on the right hand side of the Figure.

Cross Dataset Recombination. We also show more results for factor recombination across two different models trained on different datasets. In Figure 4.30, we combine inferred object components from a model trained CLEVR images and components from a model trained on CLEVR Toy images. Our method enables novel recombinations of inferred components from two different models.

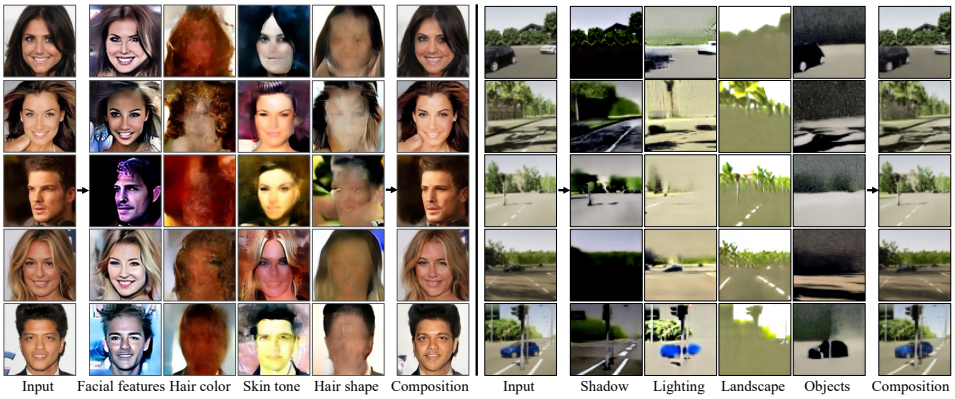

Figure 4.12: **Global Factor Decomposition.** Global factor decomposition and composition results on CelebA-HQ and Virtual KITTI 2. Note that we name inferred concepts for easier understanding.

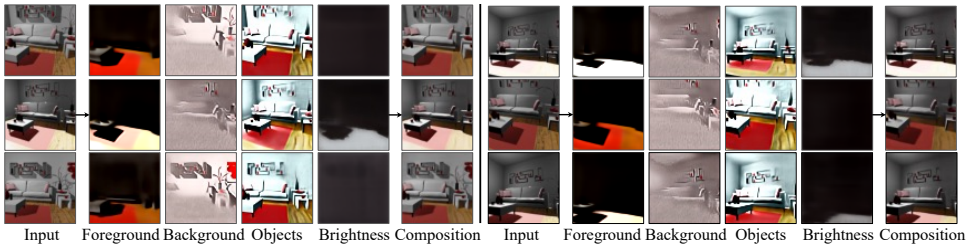

Figure 4.13: **Global Factor Decomposition.** Global factor decomposition and composition results on Falcor3D. Note that we name inferred concepts for easier understanding.

4.7.2. ADDITIONAL EXPERIMENTS

Impact of the Number of Components K. We provide qualitative comparisons on the number of components K used to train our models in Figure 4.17 and Figure 4.18.

Decomposition Comparisons. We provide qualitative comparisons of decomposed concepts in Figure 4.19 and Figure 4.21.

Factor Semantics. To visualize the impact of each decomposed factor, in Figure 4.22, we present composition results produced by incrementally adding components. On the left-hand side, we show the factors discovered for each input image. On the right-hand side, we iteratively add one factor to our latent vector subset and generate the composition results. We see that composition images steadily approach the original input image with the addition of each component. We provide similar additive composition results on the CLEVR dataset in Figure 4.23. Our method can iteratively incorporate each object represented by the learned local factors until it reconstructs the original image's object setup.

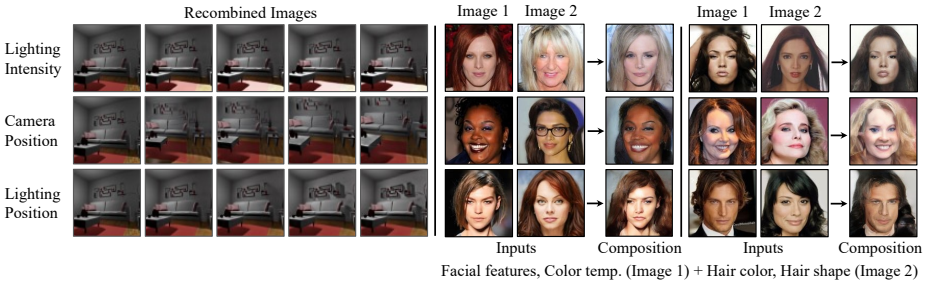

Figure 4.14: **Global Factor Recombination.** Recombination of inferred factors on Falcor3D and CelebA-HQ datasets. In Falcor3D (**Left**), we show image variations by varying inferred factors such as lighting intensity. In CelebA-HQ (**Right**), we recombine factors from two different inputs to generate novel face combinations.

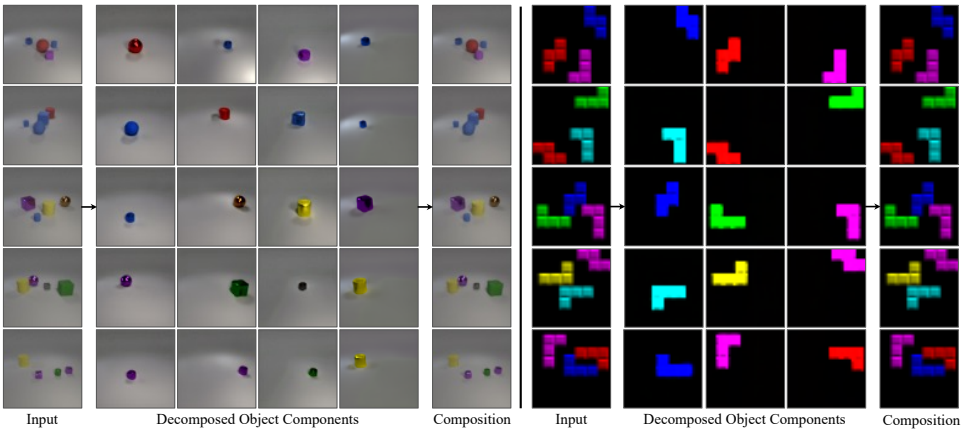

Figure 4.15: **Local Factor Decomposition.** Object-level decompositions results on CLEVR (**left**) and Tetris (**right**).

Systematic Selection of Latent Set Size. As a proxy for determining the optimal number of components for decomposition, we conduct reconstruction training by employing a weighted combination of K components, where K is sufficiently large and the weights are learned, rather than simply averaging K components. Subsequently, we utilize the weight values to identify some K' components that were less significant, indicated by their lower weights. The remaining $K - K'$ components may offer a more suitable fit for the dataset. In Figure 4.24, we used $K = 6$ and found that model learns to differentiate the importance of each component.

One-Shot Decomposition with Liu et al.. We experiment with using the method from Liu et al. 2023 [4] on a single training image to decompose CLEVR. As shown in Figure 4.25, since the method only optimizes the word embedding in the text encoder

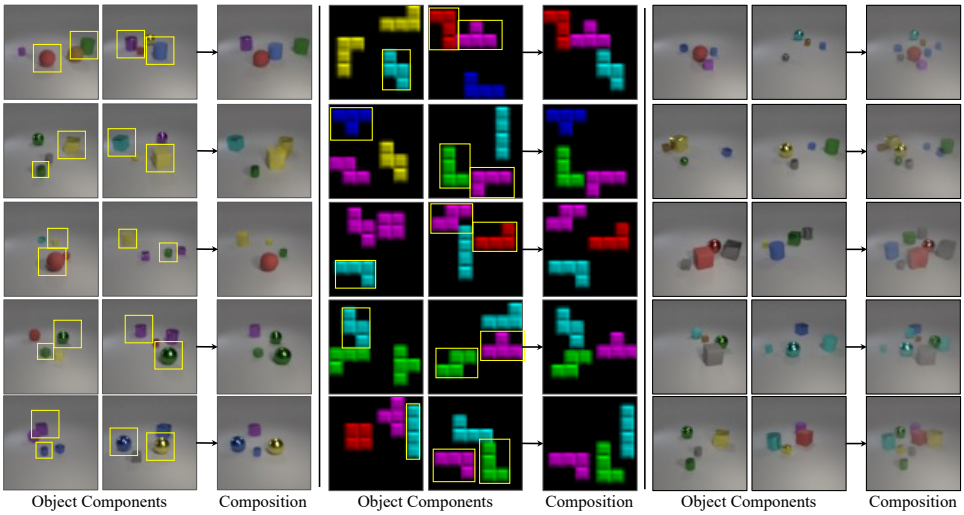

Figure 4.16: **Local Factor Recombination.** Recombination results using object-level factors from different images.

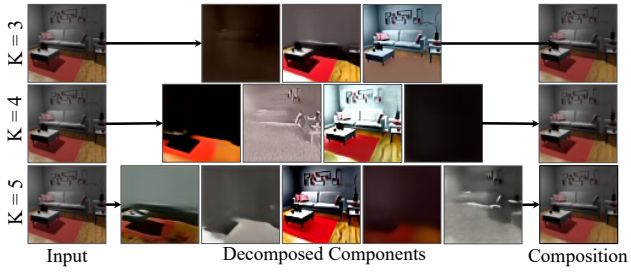

Figure 4.17: Decomp Diffusion trained on Falcor3D dataset with varying number of components $K = 3, 4,$ and 5

without updating the U-net, it does not generate objects that look similar to the training set. This suggests that the pretrained Stable Diffusion model does not always give faithful priors for factor representation learning tasks.

Decomposition with Pretrained Stable Diffusion We test a variant of our approach with pretrained Stable Diffusion without fine-tuning on the KITTI and CLEVR datasets, shown in 4.26. We can see that just using the pretrained model did not help find meaningful factors.

Impact of Latent Encoder Depth To see how the latent encoder design impacts decomposition performance, we tested decomposition on VKITTI using different encoder depths. Specifically, we experimented with an encoder of depth 1, , 1 residual block and convolution layer, as well as depth 2, depth 3 (the default value we used in

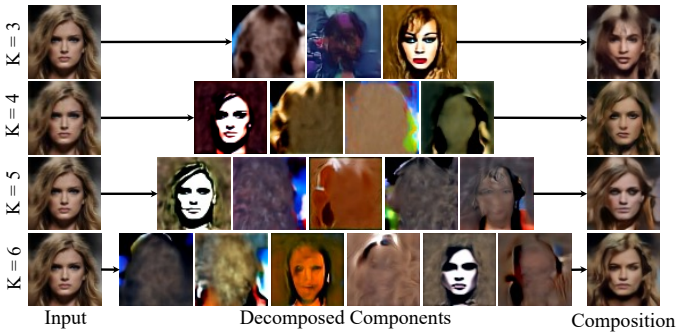

Figure 4.18: Decomp Diffusion trained on CelebA-HQ with varying number of components $K = 3, 4, 5$, and 6

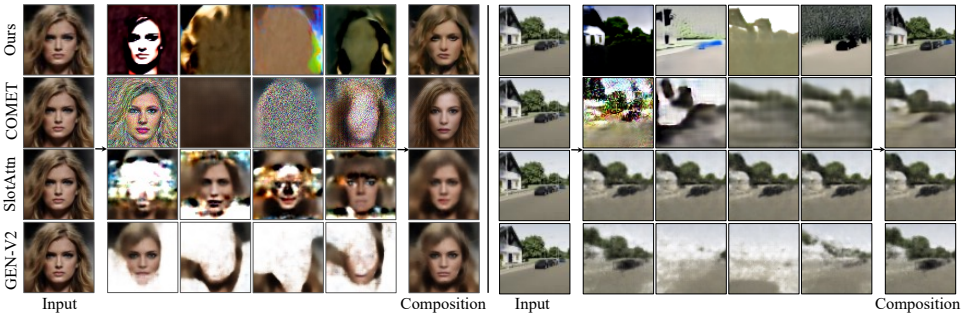

Figure 4.19: **Qualitative comparisons on CelebA-HQ and VKITTI datasets**. Decomposition results on CelebA-HQ (**Left**) and Virtual KITTI 2 (**Right**) on benchmark object representation methods. Compared to our method, COMET generates noisy components and less accurate reconstructions. SlotAttention may produce identical components, and it and GENESIS-V2 cannot disentangle global-level concepts.

the main paper), and depth 5, with results shown in Figure 4.27. We demonstrate that our method is not sensitive to encoder depth changes, as the encoders with different depths learn similar decomposed factors, including shadows, backgrounds, etc.

4.7.3. MODEL DETAILS

We used the standard U-Net architecture from [17] as our diffusion model. To condition on each inferred latent z_k, we concatenate the time embedding with encoded latent z_k, and use that as our input conditioning. In our implementation, we use the same embedding dimension for both time embedding and latent representations. Specifically, we use 256, 256, and 16 as the embedding dimension for both timesteps and latent representations for CelebA-HQ, Virtual KITTI 2, and

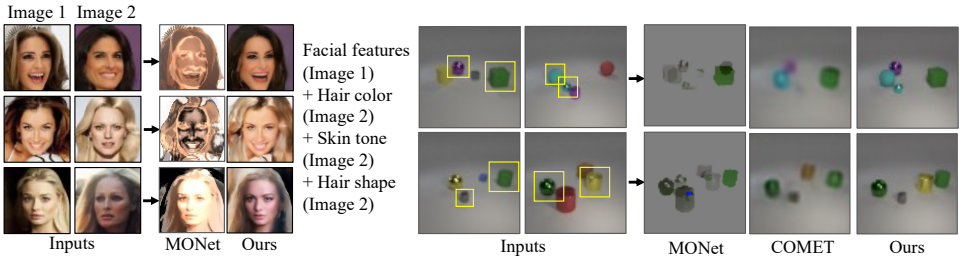

Figure 4.20: **Recombination comparisons on CelebA-HQ and CLEVR with MONet**. We further compare with MONet on recombination. Our method outperforms MONet by generating correct recombinations results.

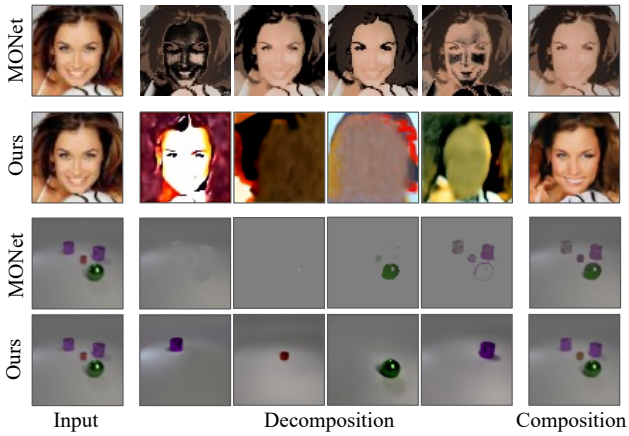

Figure 4.21: **Decomposition comparisons on CelebA-HQ and CLEVR datasets**. We provide qualitative comparisons on decomposition between MONet and our method. Our method can decompose images into factors that are more visually diverse and meaningful, while MONet may fail to disentangle factors.

Falcor3D, respectively. For datasets CLEVR, CLEVR Toy, and Tetris, we use an embedding dimension of 64.

To infer latents, we use a ResNet encoder with hidden dimension of 64 for Falcor3D, CelebA-HQ, Virtual KITTI 2, and Tetris, and hidden dimension of 128 for CLEVR and CLEVR Toy. In the encoder, we first process images using 3 ResNet Blocks with kernel size 3×3. We downsample images between each ResBlock and double the channel dimension. Finally, we flatten the processed residual features and map them to latent vectors of a desired embedding dimension through a linear layer.

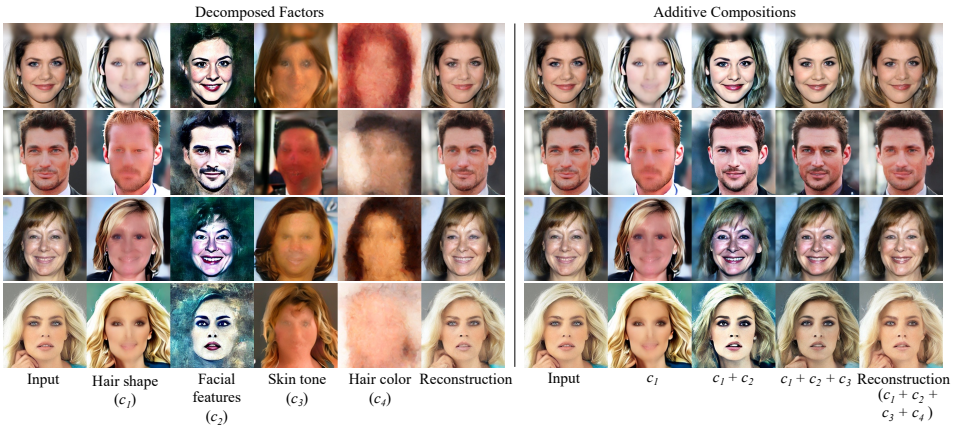

Figure 4.22: **Additive Factors Composition on CelebA-HQ.** On the left, we show decomposed components on CelebA-HQ images with inferred labels. On the right, we present compositions generated by adding one factor at a time to observe the information learned by each component.

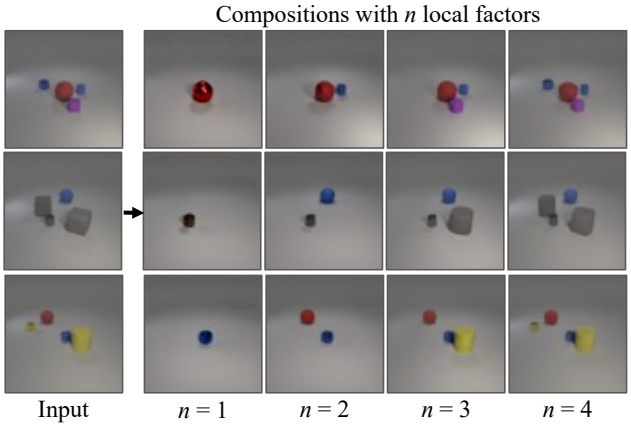

Figure 4.23: **Additive Factors Composition on CLEVR.** We demonstrate that each decomposed object factor can be additively composed to reconstruct the original input image.

4.7.4. EXPERIMENT DETAILS

In this section, we first provide dataset details for evaluation and then describe training details for our baseline methods. Finally, we present training and inference details of our method.

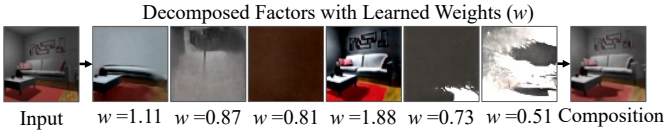

Figure 4.24: **Systematic Selection of Latent Set Size.** We can optionally learn weights for latent components during training. This approach is helpful for automatically choosing the number of components, as we can remove the most insignificant latent components based on their weights.

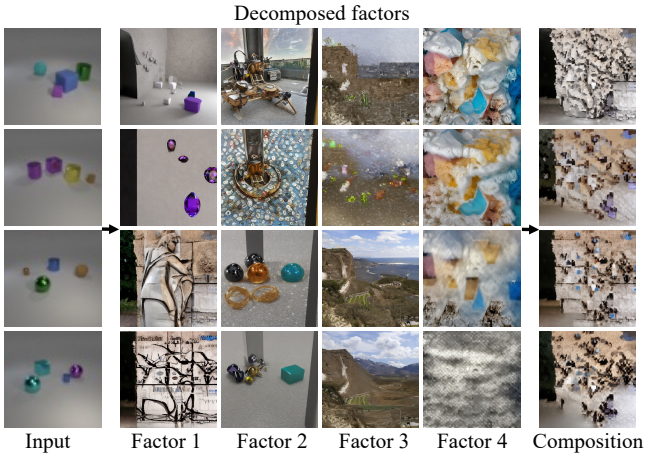

Figure 4.25: **One-Shot Decomposition using [53].** The method fails to decompose objects in the input training image.

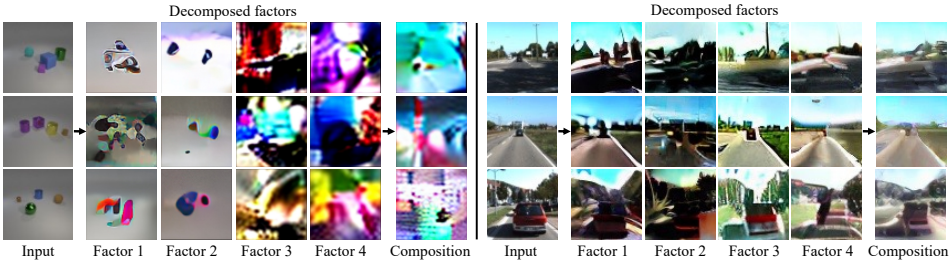

Figure 4.26: **Decomposition with Pretrained Stable Diffusion.** We find that applying our approach with pre-trained Stable Diffusion model doesn't not help find meaningful factors on both CLEVR and KITTI datasets.

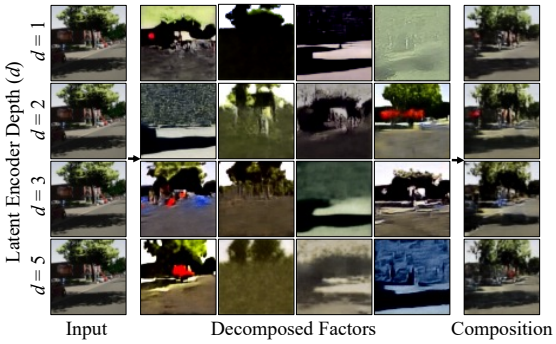

Figure 4.27: **Impact of latent encoder depth on VKITTI.** Encoders with different depths, denoted as d, can learn similar decomposed factors, including shadows, background, etc.

Dataset	Size
CLEVR	10K
CLEVR Toy	10K
CelebA-HQ	30K
Anime	30K
Tetris	10K
Falcor3D	233K
KITTI	8K
Virtual KITTI 2	21K

Table 4.4: Training dataset sizes.

DATASET DETAILS

Our training approach varies depending on the dataset used. Specifically, we utilize a resolution of 32×32 for Tetris images, while for other datasets, we use 64×64 images. The size of our training dataset is presented in Table 4.4 and typically includes all available images unless specified otherwise.

Anime. [63] When creating the multi-modal faces dataset, we combined a 30,000 cropped Anime face images with 30,000 CelebA-HQ images.

Tetris. [35] We used a smaller subset of 10K images in training, due to the simplicity of the dataset.

KITTI. [64] We used 8,008 images from a scenario in the the Stereo Evaluation 2012 benchmark in our training.

Virtual KITTI 2. [59] We used 21,260 images from a setting in different camera positions and weather conditions.

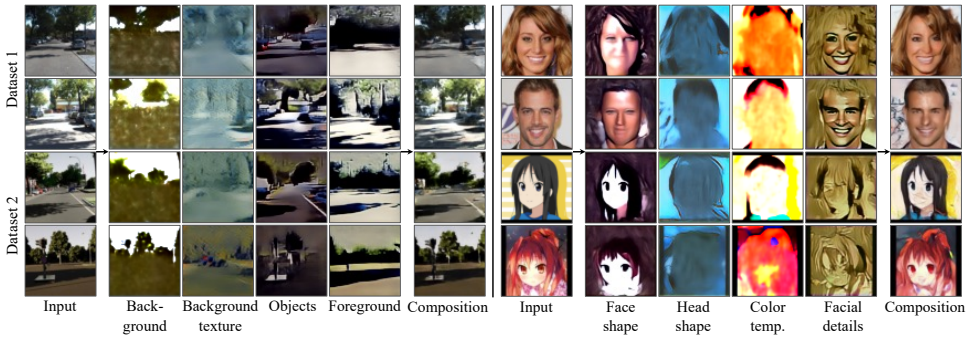

Figure 4.28: **Multi-modal Dataset Decomposition.** Multi-model decomposition and composition results on hybrid datasets such as KITTI and Virtual KITTI 2 scenes (**Left**), and CelebA-HQ and Anime faces (**Right**). The top 2 images are of the first dataset, and the bottom 2 images are of the second dataset. Inferred concepts are named for better understanding.

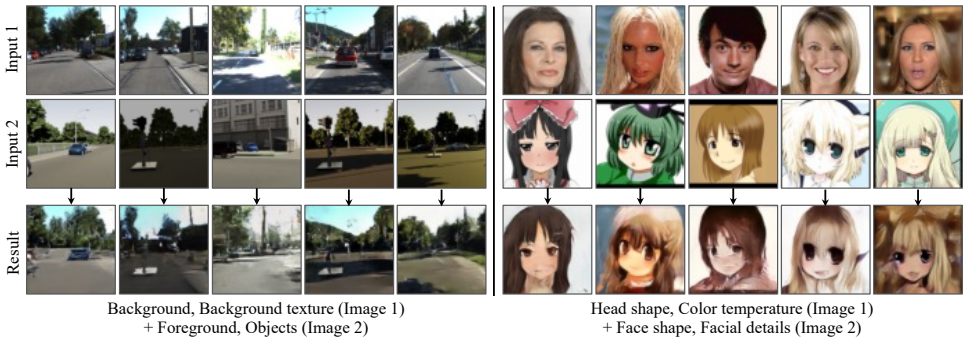

Figure 4.29: **Multi-modal Dataset Recombination.** Recombinations of inferred factors from hybrid datasets. We recombine different extracted factors to generate unique compositions of KITTI and Virtual KITTI 2 scenes (**Left**), and compositions of CelebA-HQ and Anime faces (**Right**).

BASELINES

Info-GAN [65]. We train Info-GAN using the default training settings from the official codebase at https://github.com/openai/InfoGAN.

β-VAE [30]. We utilize an unofficial codebase to train β-VAE on all datasets til the model converges. We use $\beta = 4$ and 64 for the dimension of latent $\boldsymbol{z}$. We use the codebase in https://github.com/1Konny/Beta-VAE.

MONet [9]. We use an existing codebase to train MONet models on all datasets until models converge, where we specifically use 4 slots, and 64 for the dimension of latent $\boldsymbol{z}$. We use the codebase in https://github.com/baudm/MONet-pytorch.

COMET [13]. We use the official codebase to train COMET models on various

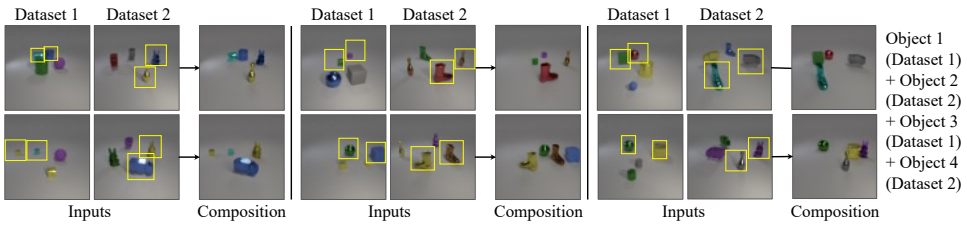

Figure 4.30: **Cross Dataset Recombination.** We further showcase our method's ability to recombine across datasets using 2 different models that train on CLEVR and CLEVR Toy, respectively. We compose inferred factors as shown in the bounding box from two different modalites to generate unseen compositions.

Model	CLEVR		CLEVR Toy	
	FID ↓	KID ↓	FID ↓	KID ↓
COMET	98.27	0.110	192.02	0.250
Ours	**75.16**	**0.086**	**52.03**	**0.052**

Table 4.5: **Cross-dataset quantitative metrics.** For evaluating cross-dataset recombination (CLEVR combined with CLEVR Toy), because there is no ground truth for recombined images, we computed FID and KID scores of generated images against the original CLEVR dataset and CLEVR Toy dataset. Our approach achieves better scores for both datasets compared to COMET, which suggests that our generations are more successful in recombining objects from the original datasets.

datasets, with a default setting that utilizes 64 as the dimension for the latent variable z. Each model is trained until convergence over a period of 100,000 iterations. We use the codebase in https://github.com/yilundu/comet.

Slot Attention [10]. We use an existing PyTorch implementation to train Slot Attention from https://github.com/evelinehong/slot-attention-pytorch .

GENESIS-V2 [66]. We train GENESIS-V2 using the default training settings from the official codebase at https://github.com/applied-ai-lab/genesis .

TRAINING DETAILS

We used standard denoising training to train our denoising networks, with 1000 diffusion steps and squared cosine beta schedule. In our implementation, the denoising network ϵ_θ is trained to directly predict the original image x_0, since we show this leads to better performance due to the similarity between our training objective and autoencoder training.

To train our diffusion model that conditions on inferred latents z_k, we first utilize the latent encoder to encode input images into features that are further split into a set of latent representations $\{z_1, \ldots, z_K\}$. For each input image, we then train our model conditioned on each decomposed latent factor z_k using standard denoising loss.

Regarding computational cost, our method uses K diffusion models, so the

computational cost is K times that of a normal diffusion model. In practice, the method is implemented as 1 denoising network that conditions on K latents, as opposed to K individual denoising networks. One could significantly reduce computational cost by fixing the earlier part of the network, since latents would only be conditioned on in the second half of the network. This would likely achieve similar results with reduced computation. In principle, we could also parallelize K forward passes to compute K score functions to reduce both training and inference time.

Each model is trained for 24 hours on an NVIDIA V100 32GB machine or an NVIDIA GeForce RTX 2080 24GB machine. We use a batch size of 32 when training.

INFERENCE DETAILS

When generating images, we use DDIM with 50 steps for faster image generation.

Decomposition. To decompose an image $\boldsymbol{x}$, we first pass it into the latent encoder Enc_θ to extract out latents $\{\boldsymbol{z}_1, \cdots, \boldsymbol{z}_K\}$. For each latent $\boldsymbol{z}_k$, we generate an image corresponding to that component by running the image generation algorithm on $\boldsymbol{z}_k$.

Reconstruction. To reconstruct an image $\boldsymbol{x}$ given latents $\{\boldsymbol{z}_1, \cdots, \boldsymbol{z}_K\}$, in the denoising process, we predict ϵ by averaging the model outputs conditioned on each individual $\boldsymbol{z}_k$. The final result is a denoised image which incorporates all inferred components, , reconstructs the image.

Recombination. To recombine images $\boldsymbol{x}$ and $\boldsymbol{x}'$, we recombine their latents $\{\boldsymbol{z}_1, \cdots, \boldsymbol{z}_K\}$ and $\{\boldsymbol{z}'_1, \cdots, \boldsymbol{z}'_K\}$. We select the desired latents from each image and condition on them in the image generation process, , predict ϵ in the denoising process by averaging the model outputs conditioned on each individual latent.

To additively combine images $\boldsymbol{x}$ and $\boldsymbol{x}'$ so that the result has all components from both images, , combining two images with 4 objects to generate an image with 8 objects, we modify the generation procedure. In the denoising process, we assign the predicted ϵ to be the average over all $2 \times K$ model outputs conditioned on individual latents in $\{\boldsymbol{z}_1, \cdots, \boldsymbol{z}_K\}$ and $\{\boldsymbol{z}'_1, \cdots, \boldsymbol{z}'_K\}$. This results in an image with all components from both input images.

REFERENCES

[1] J. Su*, N. Liu*, Y. Wang*, J. B. Tenenbaum, and Y. Du. "Compositional Image Decomposition with Diffusion Models". In: *International Conference on Machine Learning*. PMLR. 2024, 46823–46842 (* indicates equal contribution).

[2] K. R. Allen, K. A. Smith, and J. B. Tenenbaum. "Rapid trial-and-error learning with simulation supports flexible tool use and physical reasoning". In: *Proceedings of the National Academy of Sciences* 117.47 (2020), pp. 29302–29310. ISSN: 0027-8424. DOI: 10.1073/pnas.1912341117. eprint: https://www.pnas.org/content/117/47/29302.full.pdf. URL: https://www.pnas.org/content/117/47/29302.

[3] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman. "Building machines that learn and think like people". In: *Behavioral and brain sciences* 40 (2017), e253.

[4] N. Chomsky. *Aspects of the Theory of Syntax*. Cambridge: The MIT Press, 1965. URL: http://www.amazon.com/Aspects-Theory-Syntax-Noam-Chomsky/dp/0262530074.

[5] R. Vedantam, I. Fischer, J. Huang, and K. Murphy. "Generative models of visually grounded imagination". In: *ICLR*. 2018.

[6] I. Higgins, N. Sonnerat, L. Matthey, A. Pal, C. P. Burgess, M. Bosnjak, M. Shanahan, M. Botvinick, D. Hassabis, and A. Lerchner. "Scan: Learning hierarchical compositional visual concepts". In: *ICLR* (2018).

[7] K. K. Singh, U. Ojha, and Y. J. Lee. "Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 6490–6499.

[8] W. Peebles, J. Peebles, J.-Y. Zhu, A. Efros, and A. Torralba. "The hessian penalty: A weak prior for unsupervised disentanglement". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*. Springer. 2020, pp. 581–597.

[9] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner. "Monet: Unsupervised scene decomposition and representation". In: *arXiv:1901.11390* (2019).

[10] F. Locatello, D. Weissenborn, T. Unterthiner, A. Mahendran, G. Heigold, J. Uszkoreit, A. Dosovitskiy, and T. Kipf. *Object-Centric Learning with Slot Attention*. 2020. arXiv: 2006.15055 [cs.LG].

[11] T. Monnier, E. Vincent, J. Ponce, and M. Aubry. “Unsupervised layered image decomposition into object prototypes”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 8640–8650.

[12] M. Engelcke, O. Parker Jones, and I. Posner. “Genesis-v2: Inferring unordered object representations without iterative refinement”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 8085–8094.

[13] Y. Du, S. Li, Y. Sharma, B. J. Tenenbaum, and I. Mordatch. “Unsupervised Learning of Compositional Energy Concepts”. In: *Advances in Neural Information Processing Systems*. 2021.

[14] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang. “A tutorial on energy-based learning”. In: *Predicting structured data* 1.0 (2006).

[15] Y. Du and I. Mordatch. “Implicit generation and generalization in energy-based models”. In: *arXiv preprint arXiv:1903.08689* (2019).

[16] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. “Deep unsupervised learning using nonequilibrium thermodynamics”. In: *International Conference on Machine Learning*. PMLR. 2015, pp. 2256–2265.

[17] J. Ho, A. Jain, and P. Abbeel. “Denoising diffusion probabilistic models”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6840–6851.

[18] N. Liu, S. Li, Y. Du, A. Torralba, and J. B. Tenenbaum. “Compositional Visual Generation with Composable Diffusion Models”. In: *arXiv preprint arXiv:2206.01714* (2022).

[19] Y. Du, C. Durkan, R. Strudel, J. B. Tenenbaum, S. Dieleman, R. Fergus, J. Sohl-Dickstein, A. Doucet, and W. Grathwohl. “Reduce, Reuse, Recycle: Compositional Generation with Energy-Based Diffusion Models and MCMC”. In: *arXiv preprint arXiv:2302.11552* (2023).

[20] W. Feng, X. He, T.-J. Fu, V. Jampani, A. Akula, P. Narayana, S. Basu, X. E. Wang, and W. Y. Wang. “Training-Free Structured Diffusion Guidance for Compositional Text-to-Image Synthesis”. In: *arXiv preprint arXiv:2212.05032* (2022).

[21] C. Shi, H. Ni, K. Li, S. Han, M. Liang, and M. R. Min. “Exploring Compositional Visual Generation with Latent Classifier Guidance”. In: *arXiv preprint arXiv:2304.12536* (2023).

[22] Y. Cong, M. R. Min, L. E. Li, B. Rosenhahn, and M. Y. Yang. “Attribute-Centric Compositional Text-to-Image Generation”. In: *arXiv preprint arXiv:2301.01413* (2023).

[23] L. Huang, D. Chen, Y. Liu, Y. Shen, D. Zhao, and J. Zhou. “Composer: Creative and Controllable Image Synthesis with Composable Conditions”. In: *arXiv preprint arXiv:2302.09778* (2023).

[24] T. Garipov, S. De Peuter, G. Yang, V. Garg, S. Kaski, and T. Jaakkola. “Compositional sculpting of iterative generative processes”. In: *Advances in neural information processing systems* 36 (2023), pp. 12665–12702.

[25] Y. Du, S. Li, and I. Mordatch. "Compositional Visual Generation with Energy Based Models". In: *Advances in Neural Information Processing Systems*. 2020.

[26] N. Liu, S. Li, Y. Du, J. Tenenbaum, and A. Torralba. "Learning to compose visual relations". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 23166–23178.

[27] W. Nie, A. Vahdat, and A. Anandkumar. "Controllable and compositional generation with latent-space energy-based models". In: *Advances in Neural Information Processing Systems* 34 (2021).

[28] Z. Wang, L. Gui, J. Negrea, and V. Veitch. "Concept Algebra for Text-Controlled Vision Models". In: *arXiv preprint arXiv:2302.03693* (2023).

[29] Y. Du, K. A. Smith, T. Ullman, J. B. Tenenbaum, and J. Wu. "Unsupervised Discovery of 3D Physical Objects". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=lf7st0bJIA5.

[30] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. "Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". In: *ICLR*. 2017.

[31] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner. "Understanding disentangling in beta-VAE". In: *arXiv preprint arXiv:1804.03599* (2018).

[32] F. Locatello, M. Tschannen, S. Bauer, G. Rätsch, B. Schölkopf, and O. Bachem. *Disentangling Factors of Variation Using Few Labels*. 2020. arXiv: 1905.01258 [cs.LG].

[33] D. A. Klindt, L. Schott, Y. Sharma, I. Ustyuzhaninov, W. Brendel, M. Bethge, and D. Paiton. "Towards Nonlinear Disentanglement in Natural Data with Temporal Sparse Coding". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=EbIDjBynYJ8.

[34] K. Preechakul, N. Chatthee, S. Wizadwongsa, and S. Suwajanakorn. *Diffusion Autoencoders: Toward a Meaningful and Decodable Representation*. 2022. arXiv: 2111.15640 [cs.CV].

[35] K. Greff, R. L. Kaufmann, R. Kabra, N. Watters, C. Burgess, D. Zoran, L. Matthey, M. Botvinick, and A. Lerchner. "Multi-object representation learning with iterative variational inference". In: *arXiv preprint arXiv:1903.00450* (2019).

[36] T. Kipf, G. F. Elsayed, A. Mahendran, A. Stone, S. Sabour, G. Heigold, R. Jonschkowski, A. Dosovitskiy, and K. Greff. *Conditional Object-Centric Learning from Video*. 2022. arXiv: 2111.12594 [cs.CV].

[37] M. Seitzer, M. Horn, A. Zadaianchuk, D. Zietlow, T. Xiao, C.-J. Simon-Gabriel, T. He, Z. Zhang, B. Schölkopf, T. Brox, *et al.* "Bridging the gap to real-world object-centric learning". In: *arXiv preprint arXiv:2209.14860* (2022).

[38] Y. Wang, L. Liu, and J. Dauwels. "Slot-vae: Object-centric scene generation with slot attention". In: *International Conference on Machine Learning*. PMLR. 2023, pp. 36020–36035.

[39] J. Jiang, F. Deng, G. Singh, and S. Ahn. *Object-Centric Slot Diffusion*. 2023. arXiv: 2303.10834 [cs.CV].

[40] Z. Wu, J. Hu, W. Lu, I. Gilitschenski, and A. Garg. "SlotDiffusion: Object-Centric Generative Modeling with Diffusion Models". In: *arXiv preprint arXiv:2305.11281* (2023).

[41] S. Lee, Y. Zhang, S. Wu, and J. Wu. "Language-Informed Visual Concept Learning". In: *The Twelfth International Conference on Learning Representations*. 2023.

[42] H. Chefer, O. Lang, M. Geva, V. Polosukhin, A. Shocher, M. Irani, I. Mosseri, and L. Wolf. "The hidden language of diffusion models". In: *arXiv preprint arXiv:2306.00966* (2023).

[43] O. Avrahami, K. Aberman, O. Fried, D. Cohen-Or, and D. Lischinski. "Break-a-scene: Extracting multiple concepts from a single image". In: *SIGGRAPH Asia 2023 Conference Papers*. 2023, pp. 1–12.

[44] Z. Li, M. Cao, X. Wang, Z. Qi, M.-M. Cheng, and Y. Shan. "Photomaker: Customizing realistic human photos via stacked id embedding". In: *arXiv preprint arXiv:2312.04461* (2023).

[45] O. Avrahami, A. Hertz, Y. Vinker, M. Arar, S. Fruchter, O. Fried, D. Cohen-Or, and D. Lischinski. "The Chosen One: Consistent Characters in Text-to-Image Diffusion Models". In: *arXiv preprint arXiv:2311.10093* (2023).

[46] N. Kumari, B. Zhang, R. Zhang, E. Shechtman, and J.-Y. Zhu. "Multi-concept customization of text-to-image diffusion". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 1931–1941.

[47] Y. Wei, Y. Zhang, Z. Ji, J. Bai, L. Zhang, and W. Zuo. "Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 15943–15953.

[48] V. Shah, N. Ruiz, F. Cole, E. Lu, S. Lazebnik, Y. Li, and V. Jampani. "Ziplora: Any subject in any style by effectively merging loras". In: *arXiv preprint arXiv:2311.13600* (2023).

[49] F. Liu, Y. Liu, Y. Kong, K. Xu, L. Zhang, B. Yin, G. Hancke, and R. Lau. "Referring image segmentation using text supervision". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 22124–22134.

[50] M. Yi, Q. Cui, H. Wu, C. Yang, O. Yoshie, and H. Lu. "A Simple Framework for Text-Supervised Semantic Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2023, pp. 7071–7080.

[51] Y. Song, Z. Zhang, Z. Lin, S. Cohen, B. Price, J. Zhang, S. Y. Kim, and D. Aliaga. "Objectstitch: Object compositing with diffusion model". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 18310–18319.

[52] J. Xu, S. D. Mello, S. Liu, W. Byeon, T. Breuel, J. Kautz, and X. Wang. “GroupViT: Semantic Segmentation Emerges from Text Supervision”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 18113–18123. URL: https://api.semanticscholar.org/CorpusID:247026092.

[53] N. Liu, Y. Du, S. Li, J. B. Tenenbaum, and A. Torralba. “Unsupervised compositional concepts discovery with text-to-image generative models”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2085–2095.

[54] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. “Gans trained by a two time-scale update rule converge to a local nash equilibrium”. In: *Advances in Neural Information Processing Systems*. 2017, pp. 6626–6637.

[55] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton. “Demystifying mmd gans”. In: *arXiv preprint arXiv:1801.01401* (2018).

[56] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. “The unreasonable effectiveness of deep features as a perceptual metric”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 586–595.

[57] T. Karras, T. Aila, S. Laine, and J. Lehtinen. “Progressive growing of gans for improved quality, stability, and variation”. In: *arXiv preprint arXiv:1710.10196* (2017).

[58] W. Nie, T. Karras, A. Garg, S. Debnath, A. Patney, A. B. Patel, and A. Anandkumar. “Semi-supervised stylegan for disentanglement learning”. In: *Proceedings of the 37th International Conference on Machine Learning*. 2020, pp. 7360–7369.

[59] Y. Cabon, N. Murray, and M. Humenberger. “Virtual kitti 2”. In: *arXiv preprint arXiv:2001.10773* (2020).

[60] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick. “CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning”. In: *CVPR*. 2017.

[61] R. T. Q. Chen, X. Li, R. Grosse, and D. Duvenaud. “Isolating Sources of Disentanglement in Variational Autoencoders”. In: *Advances in Neural Information Processing Systems*. 2018.

[62] A. Hyvärinen and H. Morioka. “Unsupervised feature extraction by time-contrastive learning and nonlinear ICA”. In: *Advances in Neural Information Processing Systems*. 2016, pp. 3765–3773.

[63] G. Branwen, Anonymous, and D. Community. *Danbooru2019 Portraits: A Large-Scale Anime Head Illustration Dataset*. dataset. 2019.

[64] A. Geiger, P. Lenz, and R. Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2012.

[65] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. “InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets”. In: *NeurIPS*. 2016.

[66] M. Engelcke, O. Parker Jones, and I. Posner. “GENESIS-V2: Inferring Unordered Object Representations without Iterative Refinement”. In: *arXiv preprint arXiv:2104.09958* (2021).

5

COMPOSITIONAL SCENE UNDERSTANDING

In this chapter, we explore how compositional generative models can further be used not only to synthesize visual content but also to understand the properties of a scene given a natural image. We formulate scene understanding as an inverse generative modeling problem, where we seek to find conditional parameters of a visual generative model to best fit a given natural image. To enable this procedure to infer scene structure from images substantially different than those seen during training, we further propose to build this visual generative model compositionally from smaller models over pieces of a scene. We illustrate how this procedure enables us to infer the set of objects in a scene, enabling robust generalization to new test scenes with an increased number of objects of new shapes. We further illustrate how this enables us to infer global scene factors, likewise enabling robust generalization to new scenes. Finally, we illustrate how this approach can be directly applied to existing pretrained text-to-image generative models for zero-shot multi-object perception.

This chapter is based on the paper published in Proceedings of the 42nd International Conference on Machine Learning, PMLR 235:46823-46842, (2025) [1].

5.1. INTRODUCTION

5.1.1. BACKGROUNDS AND MOTIVATION

To understand surrounding physical scenes, human intelligence is able to learn abstract visual concepts from the physical world and compositionally reuse them [2–4]. Given an image of an object, we can then easily imagine how the object would look if it were rotated or moved in 3D world [5]. Such a generative learning mechanism is the key for us to accurately parse scenes that we have never encountered before, i.e., zero-shot scene understanding [6–8]. We are interested in equipping machines with such generalizable scene-understanding abilities by leveraging recent advances in generative models.

Conventionally, scene understanding tasks have been dominated by discriminative models that learn a direct mapping from input images to visual attributes [9–11], which, however, is demonstrated to struggle with generalizing to even slightly shifted test distributions [12–16]. In contrast, generative models have long been advocated for solving inference problems with the promise of better generalization brought by data generation modeling [17, 18]. Yet, only very recently have generative models begun to show promising results for visual inference tasks [19–21], thanks to the highly expressive modeling abilities of diffusion models [22, 23]. Despite this progress, these newly proposed generative inference approaches focus only on single-label classification tasks, and how to perform a broader range of scene understanding tasks (e.g., object discovery or multi-object classification) on scenes significantly more complex than those seen during training remains elusive.

In this chapter, we propose an *inverse generative modeling* framework that is broadly applicable across various scene understanding tasks, including those involving scenes more complex than that encountered during training. Our framework builds a visual generative model compositionally [24] from smaller generative pieces representing individual parts of a scene. During inference, to understand a scene, we aim to find the conditional parameters for a composed set of generative models that best fit a given natural image, enabling to fit more complex scenes by fitting a larger set of conditional parameters for more generative models.

In Figure 5.1, we show how our approach can be used to compositionally interpret scenes across different visual understanding tasks. In the top row of Figure 5.1, we illustrate how our approach can discover objects in a scene by predicting object positions and generalize effectivly to out-of-distribution images. For this task, the model is trained on *CLEVR* dataset with each image containing *3-5* objects, while tested on a different dataset *CLEVRTex* with *6-8* objects. The substantial difference in object number, shape, color, texture and background between the training set and the test set demonstrates the strong generalization ability of our approach. In the middle row of Figure 5.1, we demonstrate how our approach can simultaneously classify multiple facial attributes on *CelebA* dataset and likewise generalize faithfully, where the training set contains only female faces while the test set contains only male faces. Finally, in the bottom row of Figure 5.1, we show how our approach can adopt pretrained diffusion models to perform zero-shot multi-object perception task on web images without any additional training.

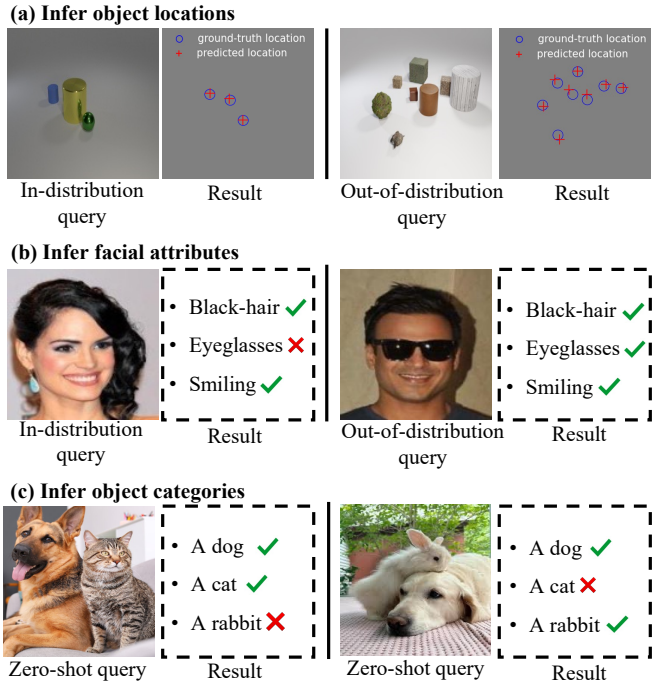

Figure 5.1: **Compositional Scene Understanding.** Our approach demonstrates strong generalization across various scene understanding tasks. For object location inference (first row), the model is trained on CLEVR images containing 3-5 objects, while the test set is CLEVRTex, which contains 6-8 objects. For multi-facial attribute inference (second row), the model is trained only on female faces from CelebA, and is tested exlusively on male faces. For object category inference (third row), we use pretrained Stable Diffusion without any additional fine-tuning, and the test set consists of multi-object natural images.

5.1.2. CHAPTER CONTRIBUTIONS

In this chapter, we contribute the following: **(1)** We propose a generic inverse generative modeling framework to tackle several scene understanding tasks such as object discovery and zero-shot multi-object perception. **(2)** We incorporate compositionality into the inverse generative modeling to enable strong generalization beyond training set. **(3)** Our approach significantly outperforms generative classifier baselines for several scene understanding tasks on both synthetic and realistic image datasets.

5.2. RELATED WORK

Generative Models for Visual Understanding. Recent work has explored applying generative models to tasks beyond visual generation, such as classification [19–21,

25–27], personalization [28–30], and segmentation [31–34]. Most relevant to our work, generative classifiers [20] leverage generative models to tackle single-label classification tasks. In contrast, our framework does not limit itself to solving single-label classification problems; instead, it demonstrates how generative models with flexible conditioning can address a broader range of visual understanding tasks, such as object discovery and zero-shot multi-object perception. More importantly, our approach composes a generative model from smaller sub-models each capturing a specific visual concept, enabling generalizing to unseen scenes that differ substantially from training set.

Compositional Generative Models. There has been significant recent progress in incorporating compositionality into generative models [24] to enable generalization beyond training distribution [35–52]. While most of these works focus on generating novel scenes, we focus on a less explored direction – inverse compositional generative modeling for scene understanding. The most similar work in this direction is UCCD [53], which requires a group of images as input to identify common concepts across image clusters. In contrast, our approach takes a single image as input, aiming to discover visual concepts that best interpret it. Furthermore, unlike UCCD relying on text-to-image generative models, our approach leverages generative models with flexible conditioning and can be applied to a wider range of visual understanding tasks.

Image Captioning. Our approach is also related to image captioning. By using pre-trained text-to-image generative models (e.g., Stable Diffusion), our model can tackle image captioning tasks like BLIP-2 [54]. However, our approach is applicable to a broader range of scene understanding tasks other than image captioning. For example, by conditioning on object coordinates, our approach can perform object discovery tasks and even enable generalization to more complex scenes (many more objects) than seen at training. This flexibility and generalizability distinguishes our approach from traditional image captioning models.

5.3. COMPOSITIONAL SCENE UNDERSTANDING THROUGH INVERSE GENERATIVE MODELING

In this section, we introduce our inverse generative modeling approach for scene understanding tasks. Given an image $\boldsymbol{x}$, we aim to infer a set of K visual components $\{\boldsymbol{c}^1, \cdots, \boldsymbol{c}^K\}$ that describe the image, where image $\boldsymbol{x}$ will often contain a larger or more complex combinations of concepts than those seen at training time. We first illustrate how we can model more complex test scenes by modeling the data generation process as a composition of a set of generative models. Next, we formulate how we can invert the generation process to infer the set of concepts that describe a given image.

5.3.1. COMPOSITIONAL GENERATIVE MODELING

In a visual domain, given a set of conditioned concepts $\{\boldsymbol{c}^1, \boldsymbol{c}^2, \cdots, \boldsymbol{c}^K\}$, we aim to construct a generative model that can accurately represent the probability distribution

$$p(\boldsymbol{x}|\boldsymbol{c}^1, \boldsymbol{c}^2, \ldots, \boldsymbol{c}^K), \tag{5.1}$$

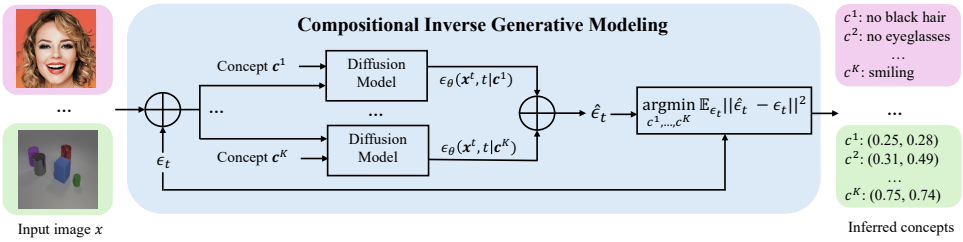

Figure 5.2: **Compositional Scene Understanding**. Our model achieves scene understanding by identifying the optimal conditioning concepts (e.g., facial attributes or object coordinates) that best interpret the input test image. Its compositional structure allows for simultaneous inference of multiple concepts and enables robust generalization to images that differ substantially from the training data.

over the space of images $\boldsymbol{x}$. The set of scenes with concepts $\{\boldsymbol{c}^1, \boldsymbol{c}^2, \cdots, \boldsymbol{c}^K\}$ can be much more complex at test time than those seen at training time, making it difficult to directly fit a generative model on the data.

One approach to model $p(\boldsymbol{x}|\boldsymbol{c}^1, \boldsymbol{c}^2, \ldots, \boldsymbol{c}^K)$ is to factorize the probability distribution [24] and approximate it as a product of simpler conditional distributions $p(\boldsymbol{x}|\boldsymbol{c}^k)$:

$$p(\boldsymbol{x}|\boldsymbol{c}^1, \ldots, \boldsymbol{c}^K) \propto \prod_{k=1}^{K} p(\boldsymbol{x}|\boldsymbol{c}^k). \tag{5.2}$$

While this approach is a biased approximation of $p(\boldsymbol{x}|\boldsymbol{c}^1, \ldots, \boldsymbol{c}^K)$, prior work [46] has found that it enables effective compositional generalization to a larger number of visual concepts. We can model each $p(\boldsymbol{x}|\boldsymbol{c}^k)$ as an EBM $e^{-E_\theta(\boldsymbol{x}|\boldsymbol{c}^k)}$. The product distribution $p(\boldsymbol{x}|\boldsymbol{c}^1, \ldots, \boldsymbol{c}^K)$ takes the form of a summation of a set of energy functions:

$$p(\boldsymbol{x}|\boldsymbol{c}^1, \ldots, \boldsymbol{c}^K) \propto e^{-\sum_{k=1}^{K} E_\theta(\boldsymbol{x}|\boldsymbol{c}^k)}. \tag{5.3}$$

To parameterize this product of EBMs, similar to [46], we can represent each EBM $E_\theta(\boldsymbol{x}|\boldsymbol{c}^k)$ using the denoising function in diffusion model $\epsilon_\theta(\boldsymbol{x}^t|\boldsymbol{c}^k, t)$ which approximately represents $\nabla_{\boldsymbol{x}} E_\theta(\boldsymbol{x}|\boldsymbol{c}^k)$. To sample from the product distribution in Equation 5.11 we can construct the composed denoising function:

$$\epsilon_\theta^{\text{comb}}(\boldsymbol{x}^t, t) = \sum_{k=1}^{K} \epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^k), \tag{5.4}$$

which approximately corresponds to $\nabla_{\boldsymbol{x}} \sum_{k=1}^{K} E_\theta(\boldsymbol{x}|\boldsymbol{c}^k)$. We can then use the composed denoising function $\epsilon_\theta^{\text{comb}}(\boldsymbol{x}^t, t)$ in the standard diffusion sampling process to approximately sample from the product distribution in Equation 5.11.

To construct the composed noise prediction model in Equation 5.10, prior work has focused on learning each denoising function $\epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^k)$ in isolation, combining denoising functions at test-time dependent on the composition needed. However, such a test-time composition of denoising functions can lead to the accumulation

of error between score functions. To more accurately model the composed score function $\epsilon_\theta^{\text{comb}}(\boldsymbol{x}^t, t)$, we directly train the composed score function with the denoising diffusion objective:

$$\begin{aligned}\mathscr{L}_\theta &= \mathbb{E}_{\boldsymbol{x},\epsilon,t}\|\epsilon - \epsilon_\theta^{\text{comb}}(\boldsymbol{x}^t, t)\|^2 \\ &= \mathbb{E}_{\boldsymbol{x},\epsilon,t}\|\epsilon - \sum_{k=1}^{K}\epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^k)\|^2,\end{aligned} \tag{5.5}$$

where each of individual term $\epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^k)$ is parameterized by a neural network. This enables the composed denoising functions to behave more accurately together, and at test time, we can still compose additional terms of $\epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^k)$ to construct more complex scenes. We provide an overview of this training approach in Algorithm 3.

5.3.2. COMPOSITIONAL SCENE UNDERSTANDING

Given a generative model $p(\boldsymbol{x}|\boldsymbol{c}^1, \boldsymbol{c}^2, \ldots, \boldsymbol{c}^K)$, we can then formulate scene understanding given an image $\boldsymbol{x}$ as an inverse problem of finding parameters of the model that explain the image. Concretely, we seek to find a set of visual concepts $\hat{\boldsymbol{c}}^1, \ldots, \hat{\boldsymbol{c}}^K$ that maximize the log-likelihood of the observed image:

$$\hat{\boldsymbol{c}}^1, \ldots, \hat{\boldsymbol{c}}^K = \underset{\boldsymbol{c}^1,\ldots,\boldsymbol{c}^K}{\arg\max} \log p(\boldsymbol{x}|\boldsymbol{c}^1, \boldsymbol{c}^2, \ldots, \boldsymbol{c}^K). \tag{5.6}$$

The inferred set of concepts $\boldsymbol{c}^k$ then corresponds to a description of the scene, where individual concepts can flexibly describe individual objects of the scene as well as global features.

We can approximate the optimization of likelihood in Equation 5.6 in diffusion models through the variational lower bound. The variational bound corresponds to a weighted form of the objective:

$$\hat{\boldsymbol{c}}^1, \ldots, \hat{\boldsymbol{c}}^K = \underset{\boldsymbol{c}^1,\ldots,\boldsymbol{c}^K}{\arg\min} \mathbb{E}_{\epsilon,t}\|\epsilon - \epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^1, \ldots, \boldsymbol{c}^K)\|^2,$$

where similar to prior work, we can approximate the likelihood by ignore the weighting terms on each loss [19]. Thus, for a given image, we can optimize for a set of visual concepts by minimizing the above objective.

We can then use the approach discussed in Section 5.3.1 to directly parameterize denoising functions for more complex scenes with more concepts by optimizing Equation 5.10, leading to the optimization objective of:

$$\hat{\boldsymbol{c}}^1, \ldots, \hat{\boldsymbol{c}}^K = \underset{\boldsymbol{c}^1,\ldots,\boldsymbol{c}^K}{\arg\min} \mathbb{E}_{\epsilon,t}\|\epsilon - \sum_{k=1}^{K}\epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^k)\|^2, \tag{5.7}$$

where we use a set of N samples with different sampled timesteps and noise to estimate the objective.

Optimizing Visual Concepts. We can solve Equation 5.7 with different optimization algorithms, depending on the concepts $\boldsymbol{c}^k$ are discrete or continuous in specific

Algorithm 3 Training Algorithm

Input: data distribution p_D, denoising model ϵ_θ, λ
while not converged **do**
  $(\boldsymbol{x}_0, \boldsymbol{c}^1, \boldsymbol{c}^2, ..., \boldsymbol{c}^K) \sim p_D$
  ▷ *Compute denoising direction*
  $\epsilon \sim \mathcal{N}(0,1), t \sim \text{Unif}(\{1,\dots,T\})$
  $\boldsymbol{x}^t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$
  $\Delta\theta \leftarrow \nabla_\theta \|\epsilon - \sum_{k=1}^K \epsilon_\theta(\boldsymbol{x}^t, t, \boldsymbol{c}^k)\|^2$
  $\theta \leftarrow \theta - \lambda\Delta\theta$
end while
return ϵ_θ

Algorithm 4 Discrete Concept Inference Algorithm

Require: an image $\boldsymbol{x}$, trained denoising model ϵ_θ
Determine all possible M^K concept configurations $\boldsymbol{c}_{tuple} = \{(\boldsymbol{c}^1, \boldsymbol{c}^2, ..., \boldsymbol{c}^K) | \boldsymbol{c}^k \in \{\ell_1^k, \ell_2^k, ..., \ell_M^k\}\}$
▷ *Evaluate denoising error for each configuration*
Initialize a denoising error list $\mathbf{E}$ = zeros(M^K)
for $n = 1,\dots,N_{\text{sample}}$ **do**
  $\epsilon_n \sim \mathcal{N}(0,1), t_n \sim \text{Unif}(\{1,\dots,T\})$
  $\boldsymbol{x}^{t_n} = \sqrt{\bar{\alpha}_t}\boldsymbol{x} + \sqrt{1-\bar{\alpha}_t}\epsilon_n$
  for $j = 1,\dots,M^K$ **do**
    $\mathbf{E}[j]$ += $\|\epsilon_n - \sum_{k=1}^K \epsilon_\theta(\boldsymbol{x}^{t_n}, t_n, \boldsymbol{c}^k_{tuple}[j])\|^2$
  end for
end for
▷ *Select the configuration with lowest denoising error*
$\hat{j} = \arg\min_{j\in\{1,\dots,M^K\}} \frac{1}{N}\mathbf{E}[j]$
return $\boldsymbol{c}_{tuple}[\hat{j}]$

visual understanding tasks. When each visual concept $\boldsymbol{c}^k \in \{\ell_1^k, \ell_2^k, ..., \ell_M^k\}$ is a discrete variable with a finite set of possibilities, we can directly optimize Equation 5.7 by enumerating through each possible configuration of $\boldsymbol{c}^k$ and evaluating the average denoising error. We illustrate this optimization in Algorithm 4. To scale to a large number of discrete concepts and reduce inference time, we further propose a gradient-based search method in Algorithm 7. In contrast, when $\boldsymbol{c}^k$ is continuous, such as when they describe the locations of objects in the scene, optimization is substantially more complex, as exhaustive search is not feasible and gradient-based optimization is easily susceptible to local minima. We describe more complex algorithms for inference when dealing with this setting in Section 5.3.3.

Inferring Number of Visual Concepts. In many scene understanding tasks, it is difficult to know beforehand the number of visual concepts K in the scene. For

instance, in an object discovery task, the number of concepts (objects) may differ from one image to another. To determine the number of concepts K for a given test scene before inferring concept parameters, we can find a number $\hat{K}$ that maximize the log-likelihood of the test image:

$$\hat{K} = \underset{K \in [K_{min}, K_{max}]}{\arg\max} \left\{ \max_{\boldsymbol{c}^1, \ldots, \boldsymbol{c}^K} \log p(\boldsymbol{x} | \boldsymbol{c}^1, \ldots, \boldsymbol{c}^K) \right\}, \tag{5.8}$$

where K_{min} and K_{max} are the minimal and maximal limit of K. Similar to Equation 5.6-Equation 5.7, we can approximate the log-likelihood in Equation 5.8 with variational lower bound and parameterize the denoising function with a composition of multiple functions for generalization purpose:

$$\hat{K} = \underset{K \in [K_{min}, K_{max}]}{\arg\min} \left\{ \min_{\boldsymbol{c}^1, \ldots, \boldsymbol{c}^K} \mathbb{E}_{\epsilon, t} \| \epsilon - \sum_{k=1}^{K} \epsilon_\theta(\boldsymbol{x}^t, t | \boldsymbol{c}^k) \|^2 \right\}. \tag{5.9}$$

5

In Figure 5.3, we visually illustrate how $\hat{K}$ can be determined in object discovery tasks by solving Equation 5.9. We can see that the ground truth number consistently yields the lowest average denoising error (highest likelihood), demonstrating the effectiveness of our concept number determination approach. We illustrate the object location visualization in Figure 5.12 and the algorithm for determining the number of concepts in Algorithm 6 in the Appendix.

Overall, our proposed inverse generative modeling (IGM) framework significantly broadens the applicability of generative models to visual understanding tasks with several key elements: (1) flexible conditioning enables the inference of continuous concepts beyond class labels; (2) compositional modeling supports simultaneous multi-concept inference besides single-concept inference; (3) compositionality allows generalization to scenes substantially different from training data; and (4) the inference algorithm is applicable to both domain-trained diffusion models and generic pretrained diffusion models. An overview of our proposed inverse generative modeling approach is illustrated in Figure 5.2. In Section 5.4.3, we demonstrate how our model can adopt pretrained text-to-image generative models like Stable Diffusion to solve zero-shot multi-object perception tasks without requiring any additional training.

5.3.3. CONTINUOUS VISUAL CONCEPT INFERENCE

In Section 5.3.2, we aim to infer a set of concepts that best describe a given image by optimizing Equation 5.7. When concepts $\boldsymbol{c}^k$ are continuous, however, the optimization using gradient descent faces several practical challenges. First, the potential non-convexity of the objective function, due to the neural network parametrization of $\epsilon_\theta(\boldsymbol{x}^t, t | \boldsymbol{c}^k)$, can cause $\boldsymbol{c}^k$ to converge to local minima, resulting in substantial deviation from the optimal solution. Second, evaluating the expectation term in Equation 5.7 at every gradient descent step incurs prohibitively high sample complexity with respect to ϵ_n and t_n, as well as significant computational complexity for evaluating $\epsilon_\theta(\boldsymbol{x}^t, t | \boldsymbol{c}^k)$. We propose improved strategies on top of gradient descent to overcome these challenges as illustrated below and outlined in Algorithm 5.

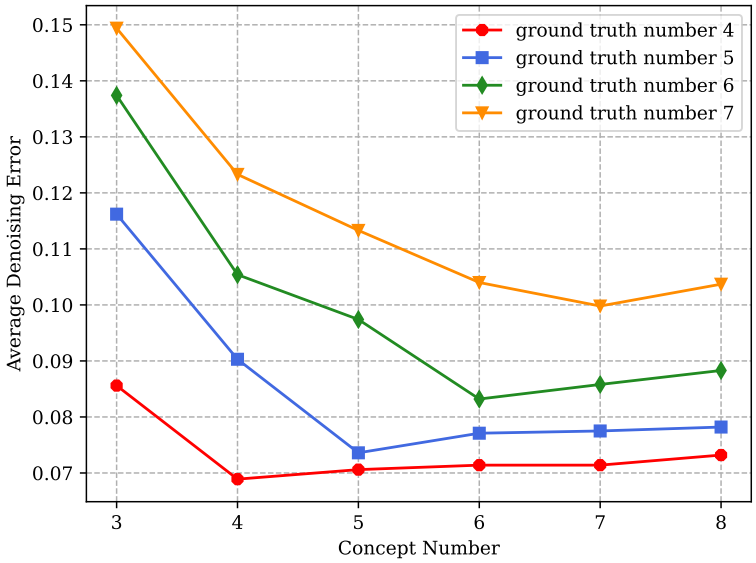

Figure 5.3: **Concept Number Inference.** Illustration of object number inference on CLEVR. Given a test image, our model evaluates each number $K \in \{3,..,8\}$ respectively by using K objects to fit the image and obtain corresponding denoising errors. Out of the potential options $K \in \{3,,..,8\}$, our model determines the one with the lowest denoising error as object number, which turns out to be consistent with the ground truth number.

Effective Concept Initialization. To more effectively prevent $\boldsymbol{c}^k$ from converging to local minima, we propose initializing $\boldsymbol{c}^k$ with multiple random starting points, denoted as $\boldsymbol{c}_1^k, \boldsymbol{c}_2^k, ..., \boldsymbol{c}_R^k$, and maintaining corresponding updates throughout the optimization process. After every few optimization steps, we terminate paths with low log-likelihood values, as the optimal $\boldsymbol{c}^k$ is expected to yield a high likelihood. This process ultimately converges to a single optimal configuration of $\boldsymbol{c}^k$ with the highest likelihood. Empirically, we find that this initialization strategy improves the algorithm's ability to escape local minima, thereby significantly enhancing scene understanding accuracy, as demonstrated in the ablation study in Sec 5.4.1 and Table 5.4.

Efficient Concept Optimization. To address the sample and computation complexity associated with evaluating the expectation term in Equation 5.7, we propose leveraging stochastic gradient descent (SGD) for optimization. This approach requires a single sample of ϵ_n and t_n at each optimization step to update the concepts $\boldsymbol{c}^k$. As a result, the sample complexity is reduced from N to 1 per iteration, and $\epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^k)$ needs to be evaluated only once per iteration in stead of N times. This significantly accelerates the inference speed.

Algorithm 5 Continuous Concept Inference Algorithm

Require: an image $\boldsymbol{x}$, trained denoising model ϵ_θ
▷ *Initialize multiple (R) sets of concepts*
Initialize concepts $\{\boldsymbol{c}_r^1, \boldsymbol{c}_r^2, ..., \boldsymbol{c}_r^K\}_{r=1}^R \sim \mathcal{N}(0,1)$
▷ *Run Stochastic Gradient Descent*
for $n = 1, \ldots, N_{\text{step}}$ **do**
  $\epsilon_n \sim \mathcal{N}(0,1), t_n \sim \text{Unif}(\{1, \ldots, T\})$
  $\boldsymbol{x}^{t_n} = \sqrt{\bar{\alpha}_{t_n}}\boldsymbol{x} + \sqrt{1-\bar{\alpha}_{t_n}}\epsilon_n$
  $\Delta \boldsymbol{c}_r^k \leftarrow \nabla_{\boldsymbol{c}_r^k} \|\epsilon_n - \sum_{k=1}^K \epsilon_\theta(\boldsymbol{x}^{t_n}, t_n, \boldsymbol{c}_r^k)\|^2$
end for
▷ *Evaluate denoising error for each set*
Initialize a denoising error list $\mathbf{E}$ = zeros(R)
for $i = 1, \ldots, N_{\text{sample}}$ **do**
  $\epsilon_i \sim \mathcal{N}(0,1), t_i \sim \text{Unif}(\{1, \ldots, T\})$
  $\boldsymbol{x}^{t_i} = \sqrt{\bar{\alpha}_{t_i}}\boldsymbol{x} + \sqrt{1-\bar{\alpha}_{t_i}}\epsilon_i$
  for $r = 1, \ldots, R$ **do**
    $\mathbf{E}[r]$ += $\|\epsilon_i - \sum_{k=1}^K \epsilon_\theta(\boldsymbol{x}^{t_i}, t_i, \boldsymbol{c}_r^k)\|^2$
  end for
end for
▷ *Select the set with lowest denoising error*
$\hat{r} = \arg\min_{r \in \{1,\ldots,R\}} \frac{1}{N}\mathbf{E}[r]$
return $\boldsymbol{c}_{\hat{r}}^1, \boldsymbol{c}_{\hat{r}}^2, ..., \boldsymbol{c}_{\hat{r}}^K$

Models	In-distri (3-5 objects)		Out-of-distri (6-8 objects)	
	PR ↑	EE ↓	PR ↑	EE ↓
ResNet-50 [55]	5.3%	$19.4e^{-2}$	2.9%	$19.7e^{-2}$
Slot Attention [56]	80.4%	$8.7e^{-4}$	53.3%	$1.3e^{-3}$
Generative Classifier [20]	82.2%	$6.0e^{-4}$	58.7%	$1.2e^{-3}$
Ours w/o multi-init	72.8%	$6.9e^{-4}$	68.0%	$7.8e^{-4}$
Ours with multi-init	**94.7%**	$\mathbf{1.4}e^{-4}$	**85.3%**	$\mathbf{3.5}e^{-4}$

Table 5.1: **Accuracy of Object Discovery.** Quantitative evaluation of object perception results on CLEVR for both in-distribution (3-5 objects) and out-of-distribution (6-8 objects) test settings. Perception rate (PR) and estimation error (EE) are reported. Our approach outperforms all the baselines, and the margin is especially significant for the out-of-distribution setting, demonstrating strong generalization capability.

5.4. EXPERIMENTS

In this section, we evaluate the scene understanding capabilities of our proposed approach across three different tasks. First, we consider a local factor perception task in Section 5.4.1, where the objective is to infer the center coordinates of objects. We next perform a global factor perception task to predict facial attributes from human

faces in Section 5.4.2. Finally, we demonstrate how our approach can be adapted to pretrained models for zero-shot multi-object perception without any additional training in Section 5.4.3.

5.4.1. LOCAL FACTOR PERCEPTION

We demonstrate how our approach can infer local factors, such as object coordinates, from a test image and effectively generalize to scenes containing a larger number of objects and more complex objects than those seen during training.

Dataset. We evaluate our approach on the CLEVR dataset [57], where each image is annotated with ground truth center coordinates of the objects. The training set consists of images containing 3-5 objects. To evaluate the generalization ability of our approach on out-of-distribution data, we consider two settings: (1) images from the CLEVR dataset containing 6-8 objects; (2) images from the CLEVRTex dataset containing 6-8 objects.

Baselines. We compare our approach against both discriminative and generative baselines, including ResNet-50 [55], Slot Attention (SlotAttn) [56], DINOSAUR [58], and Generative Classifier (GC) [20]. Details on how these baselines are trained for the object perception task can be found in Appendix 5.6.5.

Metrics. We evaluate the object discovery performance in terms of object perception rate and coordinate estimation error. The object perception rate measures the percentage of correctly discovered object relative to the total number of objects. To determine which objects are correctly discovered, we use Hungrian algorithm to match predicted coordinates and ground truth coordinates for all object in the scene. An object is considered successfully discovered if the mean square error (MSE) between the predicted coordinates and the ground truth coordinates is less than 0.002. The coordinate estimation error is computed by averaging the MSE of the predicted coordinates and the ground truth coordinates across all objects.

Qualitative Results. We qualitatively illustrate that our approach can infer object coordinates from test images more accurately than baseline models, as shown in Figure 5.4. Furthermore, we demonstrate how our approach can generalize to images with a larger number of objects and more complex objects in Figure 5.5. On the left of Figure 5.5, our model can successfully generalize to images with 6-8 objects despite being trained only on images containing 3-5 objects. In contrast, all baseline models predict object locations that significantly deviate from the ground truth. On the right of Figure 5.5, we highlight the faithful generalization ability of our model in even more challenging scenarios, where the test images are from a different dataset CLEVRTex featuring substantially different colors, textures and backgrounds compared to the training set. In this setting, our approach can still predict object location accurately, while baselines predict random location. Additional qualitative results are provided in Figure 5.8 and Figure 5.9.

Quantitative Results. We quantitatively compare our approach with baselines in Table 5.1. Our method and Generative Classifier demonstrate better perception performance than the discriminative baselines ResNet-50, Slot Attention and DINOSAUR, and our approach, by compositional modeling, achieves a 12.5% higher perception rate than Generative Classifier, with the margin increasing to 26.6% on the out-of-distribution

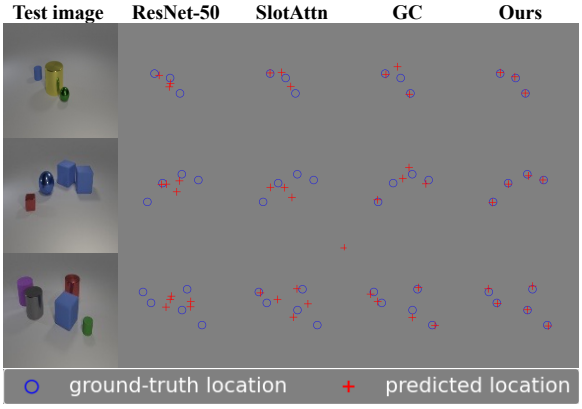

Figure 5.4: **In-distribution Object Discovery.** We train our model with CLEVR images containing 3-5 objects. During inference, given an in-distribution image (also containing 3-5 objects), our approach accurately identifies object coordinates. Compared with both determinative and generative baselines, our proposed approach demonstrates better coordinates estimation performance.

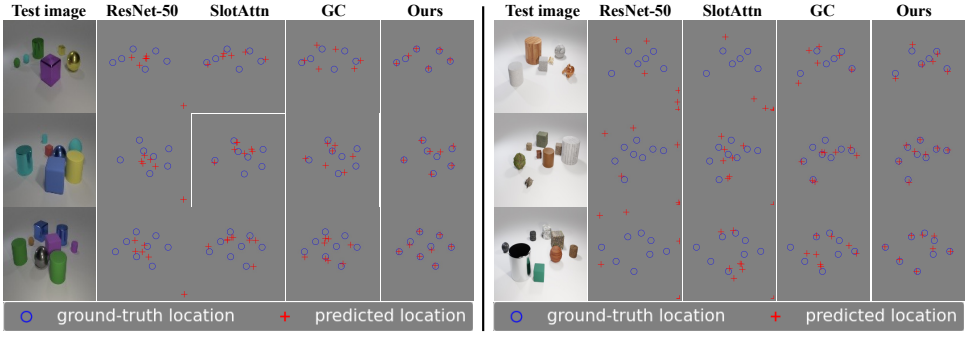

Figure 5.5: **Out-of-distribution Object Discovery.** Object perception results on out-of-distribution images: CLEVR images with 6-8 objects (**Left**) or CLEVRTex images with 6-8 objects (**Right**). Our model is trained with CLEVR images containing 3-5 objects. During inference time, given an out-of-distribution image that is substantially different from training data, our proposed approach can still infer the object positions accurately. In contrast, all baseline models predict object locations that significantly deviate from the ground truth.

tests. Meanwhile, our approach exhibits significantly lower coordinate estimation error than baselines, especially in out-of-distribution tests. Overall, these quantitative results demonstrate the strong generalization capability of our proposed approach to more complex scenes than those seen during training.

Ablation Study. We further demonstrate the effectiveness of our proposed random multiple-initialization strategy in Sec 5.3.3. As shown in Table 5.1, omitting the

random multiple-initialization strategy leads to a significant degradation in object perception performance. In this case, the algorithm often converges to local minima that substantially deviate from the ground truth coordinates, even with an increased number of optimization steps. In contrast, adopting our proposed random multiple-initialization strategy significantly improves the perception performance. Additional ablation study results can be found in Table 5.4.

5.4.2. GLOBAL FACTOR PERCEPTION

We further illustrate how our approach can infer global factors, such as facial attributes, from a test image and reliably generalize to images that differ substantially from training data.

Dataset. We evaluate our approach on the CelebA dataset [59] focusing on three attributes: Black Hair, Eyeglasses, and Smiling. The training set consists only *female faces* labeled with the these attributes, while the out-of-distribution test set comprises solely *male faces*.

Baselines. We compare our approach with both discriminative and generative approaches including ResNet-50 [55], Generative Classifier (GC) [20], and a variant of Generative Classifier. Details on how these baselines are trained for the facial feature perception task can be found in Appendix 5.6.5 .

Metrics. We evaluate facial attribute prediction performance with classification accuracy. Classification accuracy is defined as the ratio of correctly classified images to the total number of images, where an image is considered correctly classified only if all the there attributes are simultaneously predicted correctly.

Qualitative Results. We demonstrate how our approach can predict the presence of all three facial attributes in a given face image in Figure 5.6. On the left of Figure 5.6, we show that, by explicitly composing the three attributes with compositional diffusion models, our approach provides more accurate facial attribute prediction results than baselines. On the right of Figure 5.6, we further illustrate how our approach can faithfully predict facial attributes even in male faces, despite never having seen that during training, demonstrating stronger generalization compared to baselines. Additional qualitative results are provided in Figure 5.10.

Quantitative Results. We quantitatively compare our approach with baselines in Table 5.2. Our approach outperforms all baselines in classification accuracy for in-distribution images, and the performance gap becomes even pronounced for the out-of-distribution tests. This strong generalization to such non-trivial distribution shifts demonstrates the robustness of our compositional modeling strategy.

5.4.3. ZERO-SHOT MULTI-OBJECT PERCEPTION

Finally, we demonstrate that our approach can leverage pretrained diffusion models, such as Stable Diffusion [60], for zero-shot multi-object perception tasks without requiring any additional training. Specifically, we compose a set of diffusion models, each conditioned on an individual text prompt, and minimize the average denoising error with respect to the text prompts following Equation 5.7. The solution can then be obtained using Algorithm 4.

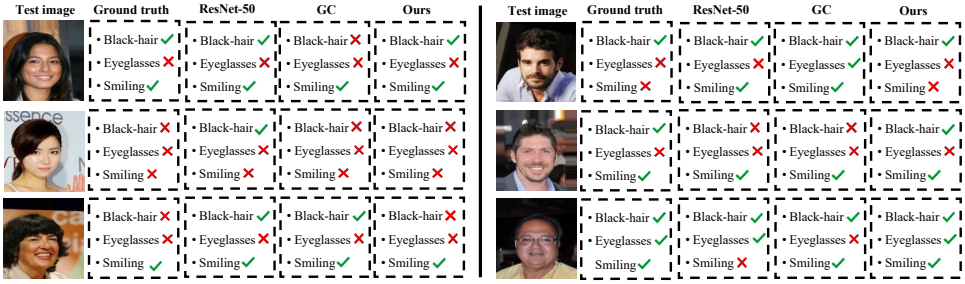

Figure 5.6: **In-Distribution and Out-of-Distribution Facial Feature Prediction.** Facial feature prediction results for in-distribution (**Left**) and out-of-distribution (**Right**) CelebA images. Our model is trained on female faces from CelebA. During inference, our model can accurately predict facial features consistent with the ground truth for both in-distribution female faces and out-of-distribution male faces.

Models	**In-distribution** (female faces)	**Out-of-distribution** (male faces)
ResNet-50	79.6%	62.2%
GC [20]	79.1%	61.7%
GC Variant	77.8%	58.1%
IGM (Ours)	**80.8%**	**65.6%**

Table 5.2: **Accuracy of Facial Feature Prediction.** Quantitative evaluation of facial feature prediction results for both in-distribution (female faces) and out-of-distribution (male faces) settings on CelebA. Our model outperforms all baseline approaches in terms of classification accuracy for the in-distribution setting and generalize even much better for the out-of-distribution setting.

In our experiment, we evaluate our model on a small dataset of 70 images, each containing two animals from the set {dog, cat, rabbit}. The prompts corresponding to these object concepts are: “a photo of dog", “a photo of cat", and “a photo of rabbit". Our model composes two diffusion models, each conditioned on any two of the prompts to interpret a given image; evaluates denoising error for the three possible prompt combinations; and selects the combination with the lowest denoising error as the optimal solution. In contrast, Diffusion Classifier (DC) [19] baseline uses a single diffusion model conditioned on compound prompts (e.g., “a photo of a dog and a cat") without explicitly modeling compositionality. The Diffusion Classifier Variant (DC variant) baseline also uses a single diffusion model but conditioned on individual object prompts, and select the two prompts with smallest denoising error as perception results. Details on Diffusion Classifier and the variant for multi-object perception can be found in Appendix 5.6.5.

As shown in Figure 5.7 and Figure 5.11, our approach can consistently recognize multiple objects in realistic images using pretrained generative models without any training. We further quantitatively compare our approach with baselines in Table

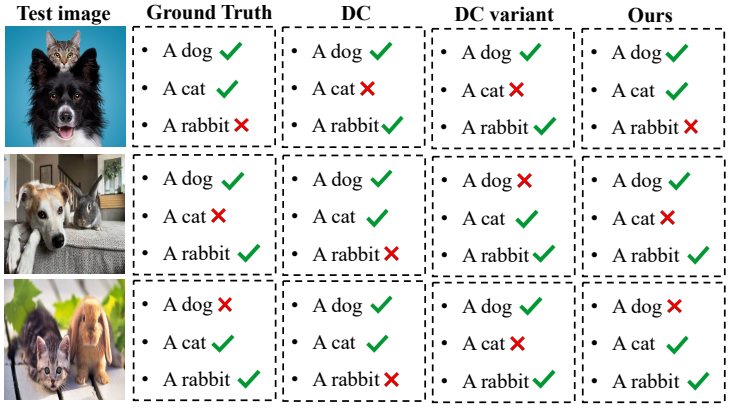

Figure 5.7: **Zero-Shot Multi-Object Perception.** Our approach can faithfully interpret given real-world images by predicting object categories that are consistent with the ground truth.

Models	Accuracy ↑
Diffusion Classifer [19]	63.8%
Diffusion Classifer Variant	61.9%
IGM (Ours)	**80.3%**

Table 5.3: **Accuracy of Zero-Shot Multi-Object Perception.** By adopting pretrained Stable Diffusion without any additional training, our compositional approach significantly outperforms Diffusion Classifier and its variant on real-world images in terms of classification accuracy for the multi-object perception task.

5.3, where our approach outperforms Diffusion Classifier by a margin of 16.5% in perception accuracy. This demonstrates that our proposed compositional generative modeling framework effectively enables multi-object scene understanding through leveraging pretrained models.

5.5. SUMMARY

Conclusions. We have presented an inverse generative modeling approach to scene understanding tasks by compositionally combining a set of generative models. We illustrate how compositionality enables the inference of visual concepts from test images that differ substantially from training data. By adopting pretrained text-to-image generative models, our model can even achieve zero-shot multi-object perception without requiring any additional training. We believe that exploring compositions of various foundation models at test time can be an promising direction to build intelligent perception systems that can generalize more effectively.

Limitations. For multiple discrete concept inference, our compositional modeling approach enumerates through each possible configuration for every concept and

evaluate denoising error for all concept combinations. This can result in long inference time when the concept number is large. To scale to a large number of concept settings, we developed a continuous approximation of our approach that allows gradient-based optimization in Algorithm 7 and Algorithm 8, thereby avoiding the exponential inference cost. Alternatively, several additional approaches can potentially significantly mitigate this computational bottleneck. One approach could be to use heuristic search algorithms on discrete values – for instance we can run beam search with a beam width of K over each attribute sequentially which can reduce time complexity from $O(M^K)$ to $O(M * K)$, making inference more efficient for large discrete spaces. Finally, since our approach allows parallel processing across the configurations, inference time can be drastically reduced given sufficient computational resources, potentially approaching the time required for a single configuration evaluation.

Another limitation is the assumption of concept independence. Our compositional generative modeling approach assumes object concept independence given the input image, enabling combinatorial generalization beyond the training distribution. However, one possible limitation of this full independence approximation is that it ignores the interaction between objects, which are crucial in many real-world scenarios. As a remedy, we could potentially learn additional models that model interactions between object components, which can also be composed to represent more complex scenes.

Social Impact. No immediate negative social impact is anticipated from our proposed approach in its current form, as we focus primarily on scene understanding tasks using standard dataset. Given the strong generalization capability to handle more complex scenes than those seen during training, our approach has the potential to benefit various fields such as autonomous driving, robot manipulation, and augmented reality, among others. Furthermore, our approach can be environmentally friendly, since our framework can leverage pretained text-to-image generative models directly for zero-shot multi-object scene understanding tasks without requiring any additional training, thereby reducing the carbon footprint.

5.6. SUPPLEMENTARY

In this appendix, we present additional details on compositional scene understanding. We illustrate additional qualitative results for various scene understanding tasks in Section 5.6.2. Next, we conduct further experiments in Section 5.6.3. Finally, we provide model details in Section 5.6.4 and experiment details in Section 5.6.5.

5.6.1. DISTRIBUTION FACTORIZATION

Assuming conditional independence among concepts $\boldsymbol{c}^1, \boldsymbol{c}^2, \ldots, \boldsymbol{c}^K$ given $\boldsymbol{x}$, the distribution $p(\boldsymbol{x}|\boldsymbol{c}^1, \boldsymbol{c}^2, \ldots, \boldsymbol{c}^K)$ can be written as:

$$p(\boldsymbol{x}|\boldsymbol{c}^1, \ldots, \boldsymbol{c}^K) \propto p(\boldsymbol{x}, \boldsymbol{c}^1, \ldots, \boldsymbol{c}^K) = p(\boldsymbol{x}) \prod_{k=1}^{K} p(\boldsymbol{c}^k|\boldsymbol{x}).$$

In this factorized form, each conditional distribution $p(\boldsymbol{c}^k|\boldsymbol{x})$ can be further approximated by the Bayes rule as $p(\boldsymbol{c}^k|\boldsymbol{x}) \propto \frac{p(\boldsymbol{x}|\boldsymbol{c}^k)}{p(\boldsymbol{x})}$, which leads to the following expression:

$$p(\boldsymbol{x}|\boldsymbol{c}^1,\dots,\boldsymbol{c}^K) \propto p(\boldsymbol{x}) \prod_{k=1}^{K} \frac{p(\boldsymbol{x}|\boldsymbol{c}^k)}{p(\boldsymbol{x})}. \tag{5.10}$$

Compared to $p(\boldsymbol{x}|\boldsymbol{c}^1,\dots,\boldsymbol{c}^K) \propto \prod_{k=1}^{K} p(\boldsymbol{x}|\boldsymbol{c}^k)$, the factorization in Equation 5.10 involves an additional unconditional term $p(\boldsymbol{x})$. Previous works such as [46, 53] explicitly model this unconditional term, while [39] assumes that $p(\boldsymbol{x})$ is uniform. In our implementation, we experiment with both approaches and selected the most effective one for each dataset (we include the unconditional term for modeling CLEVR and use only conditional terms for other datasets).

When incorporating $p(\boldsymbol{x})$ and modeling terms in Equation 5.10 as EBMs, the distribution $p(\boldsymbol{x}|\boldsymbol{c}^1,\dots,\boldsymbol{c}^K)$ then takes the form:

$$p(\boldsymbol{x}|\boldsymbol{c}^1,\dots,\boldsymbol{c}^K) \propto e^{-(E_\theta(\boldsymbol{x})+\sum_{k=1}^{K}(E_\theta(\boldsymbol{x}|\boldsymbol{c}^k)-E_\theta(\boldsymbol{x})))}, \tag{5.11}$$

and the corresponding composed denoising function approximation becomes:

$$\epsilon_\theta^{\text{comb}}(\boldsymbol{x}^t, t) = \epsilon_\theta(\boldsymbol{x}^t, t) + \sum_{k=1}^{K} (\epsilon_\theta(\boldsymbol{x}^t, t|\boldsymbol{c}^k) - \epsilon_\theta(\boldsymbol{x}^t, t)). \tag{5.12}$$

Based on this approximation, our compositional inverse generative modeling framework can train the compositional diffusion model and estimate likelihood accordingly in a way similar to Equation 5.5 and Equation 5.7.

5.6.2. ADDITIONAL QUALITATIVE RESULTS

Local Factor Perception. We illustrate additional qualitative results for the object discovery task in Figure 5.8 and Figure 5.9. Our model, despite trained only on CLEVR images with 3-5 objects, not only successfully generalizes to CLEVR images containing a larger number (6-8) of objects, but also to CLEVRTex images containing objects with substantially different colors, shapes, textures and backgrounds compared to the training images. These qualitative results demonstrate how our proposed compositional modeling by composing a set of diffusion models enables strong generalization beyond training data.

Global Factor Perception. We further illustrate additional qualitative results for the facial attribute prediction task in Figure 5.10. We train our model on female face images only from the CelebA dataset to predict facial attributes: Black Hair, Eyeglasses, and Smiling. During inference, our model is tested on male face images that differ significantly from the training images. As shown in Figure 5.10, our model can accurately predict facial attributes from male faces, demonstrating a strong ability to generalize to nontrival distribution shift.

Zero-Shot Multi-Object Perception. We illustrate additional qualitative results for the zero-shot multi-object perception task in Figure 5.11. We apply our proposed compositional inverse generative modeling framework directly to pretrained Stable

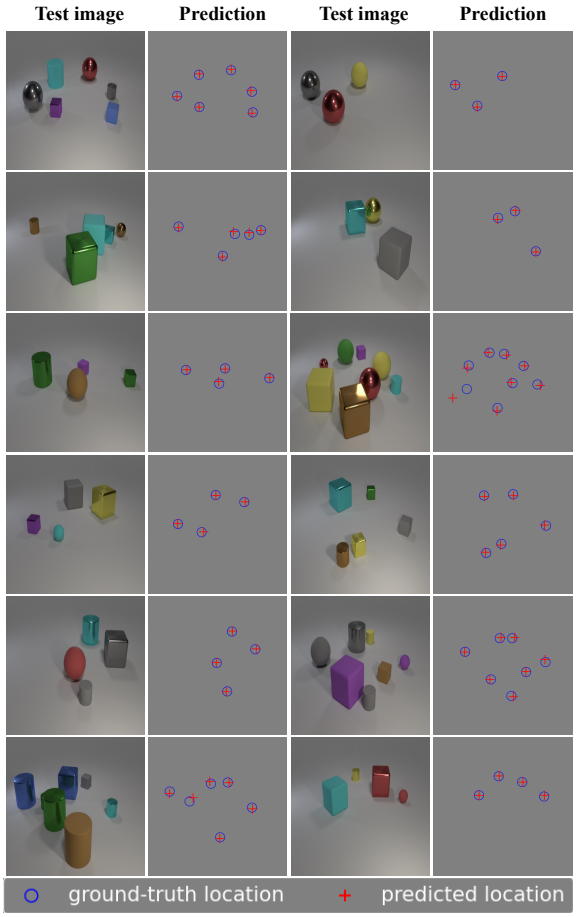

Figure 5.8: **Object Discovery Generalization.** Object discovery results on CLEVR images containing 3-8 objects. Despite trained with CLEVR images containing 3-5 objects, our model can effectively generalize to scenes with a larger number of objects.

Diffusion with any further training. Given real-world images, our model can predict object categories accurately, demonstrating an effective zero-shot scene understanding ability.

5.6.3. ADDITIONAL EXPERIMENTS

Visualization of Concept Number Inference. In Figure 5.3, we demonstrate how denoising error can serve as a criterion to select the concept number in the object discovery task. To more intuitively motivate this approach, we further illustrate visualization results in Figure 5.12 to show how inferred concepts differ from ground truth concepts when the concept number does not match the ground truth concept number. We can see that when the concept number differs from the ground truth

Figure 5.9: **Object Discovery Generalization.** Object discovery results on CLEVRTex images containing 3-8 objects. Despite trained with CLEVR images containing 3-5 objects, our model can effectively generalize to new CLEVRTex scenes containing a larger number of objects with different colors, shapes and textures.

concept number, either some concepts are missed or extra concepts are inferred. This concepts mismatch results in a large denoising error, and only the ground truth number can best fit a given image and leads to small denoising error, as reflected in Figure 5.3 and Figure 5.12. Similar to Algorithm 5, we outline the concept number inference algorithm in Algorithm 6, where we examine each possible concept number $K = K_{min}, ..., K_{max}$, figure out best concepts $\boldsymbol{c}^k$ under each K, evaluate average denoising error for each K, and select the one configuration of K with smallest average denoising error as the concept number estimate.

Multiple Random Initialization Strategy Ablation. We discussed the importance of multiple random initialization strategy for continuous concept inference on top of

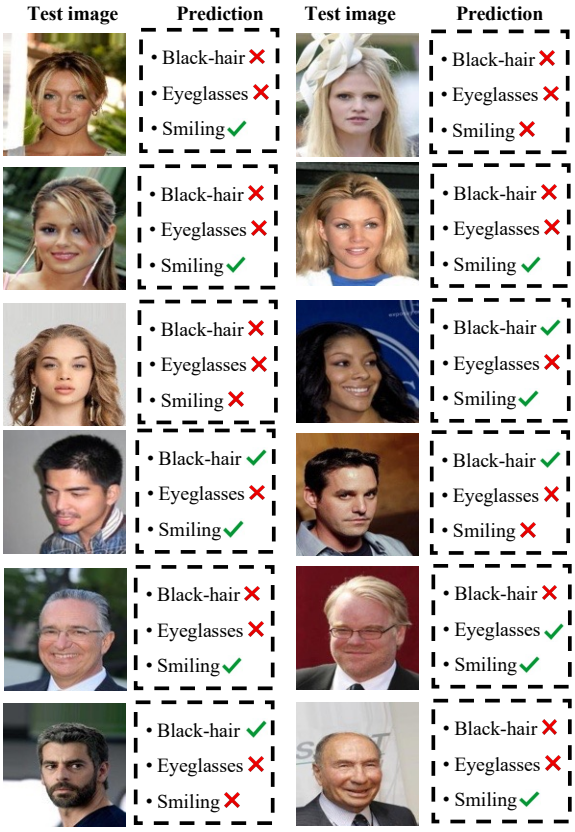

Figure 5.10: **Facial Attribute Prediction Generalization.** Facial attribute prediction results on ClebaA images. Despite trained with only female faces from CelebA, our model can effectively generalize to new scenes containing male faces.

Stochastic Gradient Descent [61] in Section 5.3.3 and we presented the results of an ablation study in Table 5.1. To further demonstrate the effectiveness of this approach, we visually illustrate how single random initialization may converge to either an optimal solution or a local minima in Figure 5.13. By employing multiple random initializations, our approach ensembles several starting points and corresponding optimization paths, achieving significant improvement over the capability to converge to the optimal solution. In Table 5.4, we further quantitatively demonstrate that as the number of random initialization starting points increases, the perception performance of our model improves.

Prompt Weighting for Multi-Concept Perception. We conducted an additional comparison to see whether naïve prompt weighting could solve the zero-shot perception task, using the Compel package. Specifically, we applied prompt weighting to the compound prompts including “a photo of a cat++, a dog, and a rabbit”, “a photo of a cat, a dog++, and a rabbit”, and “a photo of a cat, a dog, and a rabbit++”.

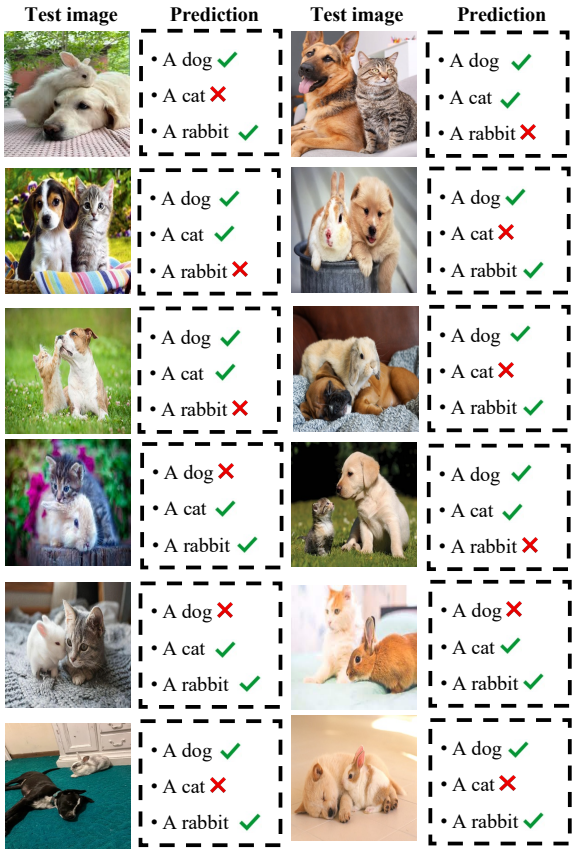

Figure 5.11: **Zero-Shot Multi-Object Perception.** Zero-shot multi-bbject perception results on natural images containg two animals from a finite set {dog, cat, rabbit}. By leveraging pretrained Stable Diffusion without any additional training, our model predicts object categories accurately.

The underlying idea is that if the image contains specific concepts (e.g., a cat and a dog), then the prompts "a photo of a cat++, a dog, and a rabbit" and "a photo of a cat, a dog++, and a rabbit" are expected to result in lower denoising error (higher likelihood) than the prompt "a photo of a cat, a dog, and a rabbit++". To determine what two objects present in the scene, we choose the two prompt weighted compound prompts that have the lowest denoising error. We report the zero-shot perception accuracy in Table 5.5. The results suggest that simple prompt weighting may not be sufficient for effective multi-concept inference in this setting.

Object Discovery on CLEVRTex. Our model has been evaluated on widely adopted datasets (CLEVR and CelebA) that are commonly used for object discovery and multi-label classification. While these datasets are relatively simple, they serve as strong benchmarks for measuring effectiveness and generalization. Nonetheless, we

Algorithm 6 Concept Number Inference Algorithm

Require: an image $\boldsymbol{x}$, trained denoising model ϵ_θ
Initialize denoising error list $\mathbf{E}$ = zeros($K_{max} - K_{min}$)
for $K = K_{min}, ...K_{max}$ **do**
  ▷ *Initialize multiple (R) groups of concepts*
  Initialize concepts $\{\boldsymbol{c}_r^1, \boldsymbol{c}_r^2, ..., \boldsymbol{c}_r^K\}_{r=1}^R \sim \mathcal{N}(0,1)$
  ▷ *Run Stochastic Gradient Descent*
  for $n = 1, \ldots, N_{\text{step}}$ **do**
    $\epsilon_n \sim \mathcal{N}(0,1), t_n \sim \text{Unif}(\{1, \ldots, T\})$
    $\boldsymbol{x}^{t_n} = \sqrt{\bar{\alpha}_{t_n}}\boldsymbol{x} + \sqrt{1-\bar{\alpha}_{t_n}}\epsilon_n$
    $\Delta \boldsymbol{c}_r^k \leftarrow \nabla_{\boldsymbol{c}_r^k} \|\epsilon_n - \sum_{k=1}^K \epsilon_\theta(\boldsymbol{x}^{t_n}, t_n, \boldsymbol{c}_r^k)\|^2$
  end for
  ▷ *Evaluate denoising error for each configuration*
  Initialize a denoising error list $\mathbf{E_R}$ = zeros(R)
  for $i = 1, \ldots, N_{\text{sample}}$ **do**
    $\epsilon_i \sim \mathcal{N}(0,1), t_i \sim \text{Unif}(\{1, \ldots, T\})$
    $\boldsymbol{x}^{t_i} = \sqrt{\bar{\alpha}_{t_i}}\boldsymbol{x} + \sqrt{1-\bar{\alpha}_{t_i}}\epsilon_i$
    for $r = 1, \ldots, R$ **do**
      $\mathbf{E_R}[r]$ += $\|\epsilon_i - \sum_{k=1}^K \epsilon_\theta(\boldsymbol{x}^{t_i}, t_i, \boldsymbol{c}_r^k)\|^2$
    end for
  end for
  ▷ *Select the group with lowest denoising error*
  $\mathbf{E}[K] = \min_{r \in \{1,...,R\}} \frac{1}{N}\mathbf{E_R}[r]$
end for
$\hat{K} = \arg\min_{K \in \{K_{min},...,K_{max}\}} \mathbf{E}[K]$
return $\hat{K}$

believe that testing our model on more complex scenes would be interesting. To take a first step towards that end, we conducted additional evaluations on ClevrTex [62], which features diverse object colors, textures, shapes, and complex backgrounds. As shown in the Table 5.6, our model outperforms all baselines in object discovery on ClevrTex, demonstrating its potential scalability to more complex scenarios.

Inference Time To evaluate inference efficiency, we conducted a comparison of runtime performance between our method and baseline models on an NVIDIA H100 GPU. The table below shows the inference time for the discrete concept inference on CelebA considering 4 attributes. As shown in Table 5.7, the inference time of our approach is comparable to the baseline model Diffusion Classifier. To further enable our model to work on a large number of concept settings (i.e., larger K), we developed a continuous approximation of our approach that allows gradient-based optimization, thereby avoiding the exponential (M^K) inference cost. The gradient-based discrete concept inference algorithm is outlined in Algorithm 7.

Specifically, to infer binary labels with gradient-based optimization, we relax the learnable binary labels to continuous parameters in the range (0, 1). These continuous

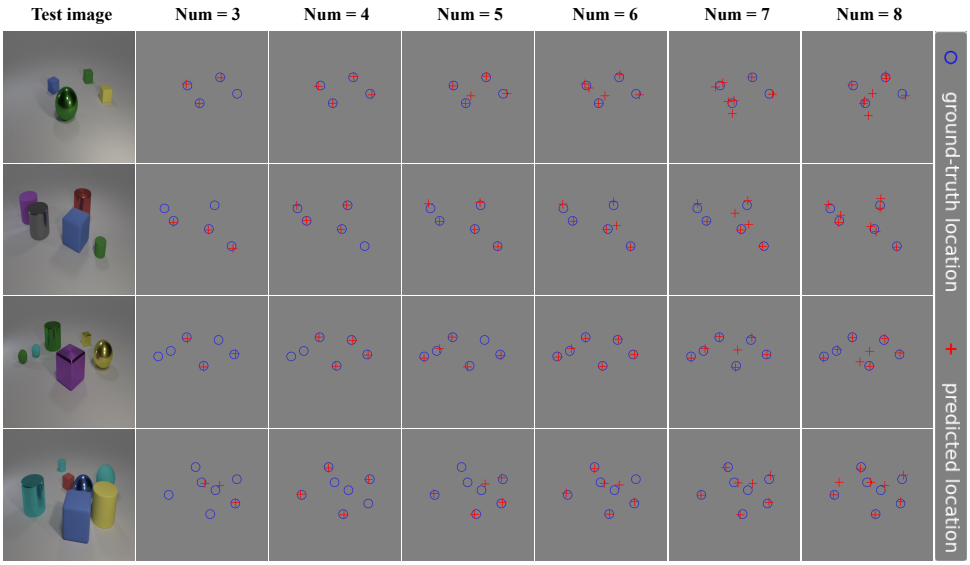

Figure 5.12: **Visualization of Object Number Inference in Object Discovery Tasks.** We train our model with CLEVR images containing 3-5 objects. During inference, given a test image, we try to use $K = 3, ..., 8$ object coordinates to fit the image through our inverse generative modeling approach following Algorithm 5 and Algorithm 6. We can see how the estimated coordinates mismatches ground truth coordinates when the object number differs from ground truth number.

Models	**In-distri (3-5 objects)**		**Out-of-distri (6-8 objects)**	
	PR ↑	EE ↓	PR ↑	EE ↓
IGM 1-init (Ours)	72.8%	$6.9e^{-4}$	68.0%	$7.8e^{-4}$
IGM 5-init (Ours)	89.6%	$2.0e^{-4}$	79.1%	$5.4e^{-4}$
IGM 10-init (Ours)	90.5%	$1.9e^{-4}$	81.6%	$4.6e^{-4}$
IGM 15-init (Ours)	92.8%	$1.6e^{-4}$	84.3%	$3.5e^{-4}$
IGM 20-init (Ours)	**94.7%**	$\mathbf{1.4}e^{-4}$	**85.3%**	$\mathbf{3.5}e^{-4}$

Table 5.4: **Ablation Study of Multiple Random Initialization Strategy.** We illustrate a quantitative evaluation of object perception results on CLEVR for both in-distribution (3-5 objects) and out-of-distribution (6-8 objects) test settings. The object coordinates are inferred using Algorithm 5 with varying numbers of random initializations for all concepts. Perception rate (PR) and estimation error (EE) are reported. The quantitative results indicate that as the number of initialization starting points increases, the perception performance improves consistently. This demonstrates that our proposed multiple random initialization strategy can help avoid convergence to local minima for continuous concept inference.

parameters are optimized using gradient descent and clamped to (0, 1) at each step to remain valid. After optimization, we decide a label is 0 if the corresponding

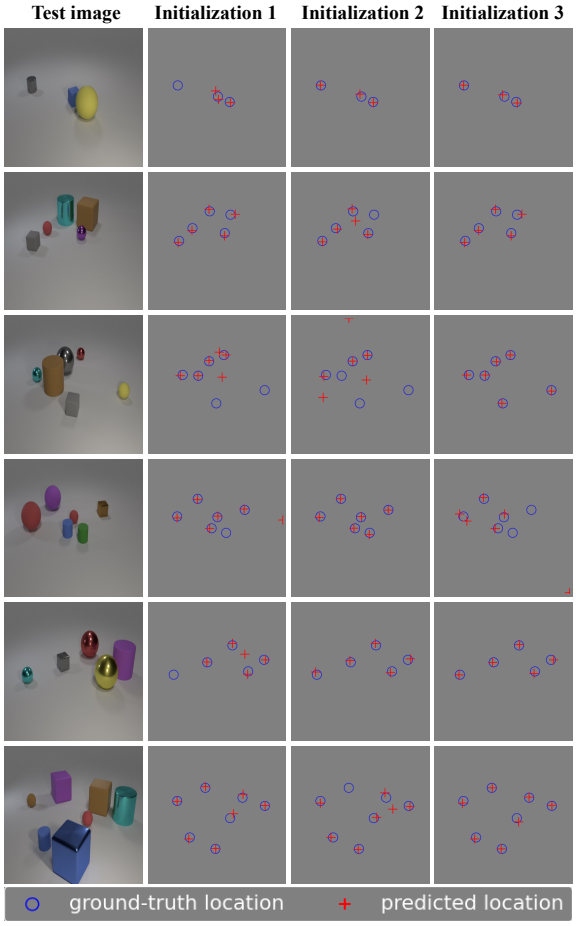

Figure 5.13: **Visualization of Random Concept Initialization in Object Discovery Tasks.** We train our model with CLEVR images containing 3-5 objects. During inference time, given a test image, we initialize object coordinates randomly and run stochastic gradient descent to iteratively refine the coordinates. Due to the non-convexity of the problem, the optimization procedure can converge either to the optimal solution or local minima, depending on initialization values. We propose to employ multiple random initializations, so that our approach can ensemble several starting points and corresponding optimization paths, achieving significant improvement over the capability to converge to the optimal solution.

optimized relaxed parameter is smaller than 0.5, otherwise the label is 1. As is shown in the table below, the continuous gradient-based approach reduces inference time significantly, making it scale linearly with the number of concepts ($O(K)$).

Additionally, we also provide the runtime of our approach and baselines on the zero-shot object perception task. Similar to the previous setting, we have developed

Models	OOD Accuracy ↑
Diffusion Classifer	63.8%
Compel	**35.2%**
Ours	80.3%

Table 5.5: **Accuracy of Zero-Shot Perception with Prompt Weighting.** We evaluate the perception accuracy of the prompt weighting approach on the animal dataset and demonstrate that our approach significantly outperforms it.

Models	**In-distribution (3-5 objects)**		**Out-of-distribution (6-8 objects)**	
	PR ↑	EE ↓	PR ↑	EE ↓
ResNet-50 [55]	3.9%	$2.0e^{-3}$	1.8%	$2.0e^{-3}$
SlotAttn [56]	41.9%	$1.5e^{-3}$	35.2%	$1.6e^{-3}$
GC [20]	69.6%	$9.8e^{-4}$	52.9%	$1.4e^{-3}$
Ours	85.2%	$5.1e^{-4}$	72.4%	$7.8e^{-4}$

Table 5.6: **Accuracy of Object Discovery.** Quantitative evaluation of object perception results on CLEVRTex for both in-distribution (3-5 objects) and out-of-distribution (6-8 objects) test settings. Perception rate (PR) and estimation error (EE) are reported. Our approach outperforms all the baselines, and the margin is especially significant for the out-of-distribution setting, demonstrating strong generalization capability.

Models	OOD Accuracy ↑	Inference Time ↓
Diffusion Classifer li2023your	51%	28.49s
IGM (Ours)	**60%**	29.10s
IGM (Ours)- continuous approx	55%	**22.15s**

Table 5.7: **Accuracy and Inference Time of Global Factor Perception.** The inference time of both our approach and Diffusion Classifer are evaluated on CelebA. To avoid exponential computation cost for discrete concept inference through enumeration, we further develop a continuous approximation of our approach through gradient-based search algorithm, which significant reduces inference time while maintains generalization performance.

a continuous approximation for the zero-shot object perception task to improve inference efficiency. The gradient-based multi-object perception algorithm is outlined in Algorithm 8.

Specifically, for each concept (e.g., “a photo of a cat”), we assign a learnable weight to its corresponding noise prediction in the compositional model. These weights are then optimized via gradient descent. After optimization, we select the top two concepts with the highest optimized weights as the predicted objects in the scene. As shown in Table 5.8 this continuous relaxation leads to a significant reduction in inference time, with the time complexity scaling linearly with the number of

Algorithm 7 Gradient-based Discrete Concept Inference Algorithm

Require: an image $\boldsymbol{x}$, trained denoising model ϵ_θ
Initialize relaxed continuous labels $\boldsymbol{l}^1, ..., \boldsymbol{l}^K \in (0,1)$
▷ *Construct psudo one-hot encoding for labels*
$\boldsymbol{c}^1 = [\boldsymbol{l}^1, 1-\boldsymbol{l}^1, 0, 0, 0..., 0, 0]$
$\boldsymbol{c}^2 = [0, 0, \boldsymbol{l}^2, 1-\boldsymbol{l}^2, 0..., 0, 0]$
.....
$\boldsymbol{c}^K = [0, 0, 0, 0, 0..., \boldsymbol{l}^K, 1-\boldsymbol{l}^K]$
▷ *Run Stochastic Gradient Descent*
for $n = 1, \ldots, N_{\text{step}}$ **do**
  $\epsilon_n \sim \mathcal{N}(0,1), t_n \sim \text{Unif}(\{1, \ldots, T\})$
  $\boldsymbol{x}^{t_n} = \sqrt{\bar{\alpha}_{t_n}}\boldsymbol{x} + \sqrt{1-\bar{\alpha}_{t_n}}\epsilon_n$
  $\boldsymbol{l}^k \leftarrow \boldsymbol{l}^k - \lambda \nabla_{\boldsymbol{l}^k} \|\epsilon_n - \sum_{k=1}^{K} \epsilon_\theta(\boldsymbol{x}^{t_n}, t_n, \boldsymbol{c}^k)\|^2$
  ▷ *Clamp* $\boldsymbol{l}^k$ *to (0, 1)*
  $\boldsymbol{l}^k \leftarrow \boldsymbol{l}^k.clamp(0,1)$
end for
if $\boldsymbol{l}^k < 0.5$ **then**
  $\boldsymbol{l}^k \leftarrow 0$
else
  $\boldsymbol{l}^k \leftarrow 1$
end if
return $\boldsymbol{l}^1, \boldsymbol{l}^2, ..., \boldsymbol{l}^K$

Algorithm 8 Gradient-based Zero-Shot Perception Algorithm

Require: an image $\boldsymbol{x}$, trained denoising model ϵ_θ, $prompt^1$ = "A photo of a cat", $prompt^2$ = "A photo of a dog", $prompt^3$ = "A photo of a rabbit"
▷ *Get Text Embeddings for Prompts*
$\boldsymbol{c}^k = \text{TextEmbedding}(prompt^k)$
Initialize concept weights $\boldsymbol{w}^1, \boldsymbol{w}^2, \boldsymbol{w}^3$
▷ *Run Stochastic Gradient Descent*
for $n = 1, \ldots, N_{\text{step}}$ **do**
  $\epsilon_n \sim \mathcal{N}(0,1), t_n \sim \text{Unif}(\{1, \ldots, T\})$
  $\boldsymbol{x}^{t_n} = \sqrt{\bar{\alpha}_{t_n}}\boldsymbol{x} + \sqrt{1-\bar{\alpha}_{t_n}}\epsilon_n$
  $\boldsymbol{w}^k \leftarrow \boldsymbol{w}^k - \lambda \nabla \|\epsilon_n - \sum_{k=1}^{K} \boldsymbol{w}^k \epsilon_\theta(\boldsymbol{x}^{t_n}, t_n, \boldsymbol{c}^k)\|^2$
end for
▷ *Select the two indices with largest weights*
indices = $top2([\boldsymbol{w}^1, \boldsymbol{w}^2, \boldsymbol{w}^3])$
return $\boldsymbol{c}^{\text{indices}}$

candidate concepts ($O(K)$).

Models	OOD Accuracy ↑	Inference Time ↓
Diffusion Classifer [19]	64%	99.44s
IGM (Ours)	**80%**	179.96s
IGM (Ours)- continuous approx	68%	**101.05s**

Table 5.8: **Accuracy and Inference Time of Zero-Shot Multi-Object Perception.** The inference time of both our approach and Diffusion Classifer are evaluated on the zero-shot multi-object perception task. To avoid exponential computation cost for discrete concept inference through enumeration, we further develop a continuous approximation of our approach through gradient-based search algorithm, which significant reduces inference time while maintains generalization performance.

5.6.4. MODEL DETAILS

We used the Unet2DConditionModel from the diffusers library as our diffusion model. We use the center coordinates in Clevr, the one-hot encoding of labels in CelebA, and CLIP embeddings of text descriptions in the animal dataset as conditionings of the Unet. For both Clevr and CelebA, we use channels [128, 256, 512, 512] for downsampling blocks. For the animal dataset, we use default Unet configuration for pretrained Stable Diffusion 2.1.

5.6.5. EXPERIMENT DETAILS

DATESET DETAILS

Object Discovery. We train our compositional generative model on 50000 CLEVR images [57], each containing 3-5 objects of varying color, shape, size and texture. We resize the training images to a resolution of 64×64. For each image, the 2-D center coordinates of objects are also available, which are of continuous value and normalized between 0 and 1 by dividing the coordinates in terms of pixel value by the image resolution. The out-of-distribution test set consist of images with 6-8 objects either from the CLEVR dataset [57], or from the CLEVRTex dataset [62].

Facial Feature Prediction. For our facial feature experiments, we trained our model with 40612 female face images from the CelebA dataset [59]. For each face image, three attributes are available including: Black Hair, Eyeglasses, and Smiling, each represented by categorical values $\{-1, 1\}$. We represent the attribute labels with one-hot encoding during training. The out-of-distribution test set consists of male faces only.

Zero-Shot Multi-Object Perception. For the zero-shot multi-object perception task, our approach directly leverages pretrained text-to-image generative models,requiring no additional training data. To evaluate the perception performance of our proposed approach in this task, we manually collected a small real-world dataset consisting of 70 random realistic images from the Internet, each containing two animals from $\{\text{dog}, \text{cat}, \text{rabbit}\}$. In line with the common practice in text-to-image generative models, the prompt corresponding to these object concepts are: “a photo of cat”, “a photo of dog”, and “a photo of rabbit”.

TRAINING DETAILS

We train a conditional latent diffusion model with latent space of 4 channels and resolution 8×8, which uses pretrained VAE to encode input images into the latent space. The latent space image is scaled with a factor of 0.18215. The denoising network adopts the Unet architecture [63] as commonly used in diffusion models that takes the latent space image as input along with label conditioning and outputs noise predictions. Specifically, the input for the denoising network is of 8×8 and the cross attention dimension is 2 (the object coordinates dimension is 2) for object discovery and 6 (the one-hot encoding of facial attributes is of dimension 6) for facial feature prediction. We use 1000 diffusion steps and linear beta schedule for training. For other hyperparameters, we use a batch size 128 and a learning rate $2e^{-5}$.

BASELINES DETAILS.

We compare our model against multiple discriminative and generative baselines. In this section, we introduce details on how these baselines are trained for scene understanding tasks considered in Section 5.4.

ResNet-50 for Object Discovery. For the object discovery tasks, the maximal number of objects in images is 5 in the training set and 8 in the test set. To enable ResNet-50 [55] to infer coordinates from images with 8 objects, we append a linear layer with input dimension 2048 and output dimension 16 on top of ResNet-50, followed by a sigmoid layer that outputs values between 0 and 1. The outputs is further reorganized into a 8×2 matrix with each row representing center coordinates of an object. We match the model output with ground truth object coordinates by minimizing the MSE loss to train the model. Since the dimension of ground truth coordinates in training data is $K \times 2$, where $3 < K < 5$, we pad additional $8 - K$ coordinates with values $(1, 1)$ representing empty coordinates (no object) to match the dimension of model output.

Slot Attention for Object Discovery. Slot Attention [56] is an unsupervised discriminative method for object discovery with strong generalization performance, which, however, only provides segmentation masks without giving the center coordinates of objects. For a fair comparison, we modify and train Slot Attention with object coordinates supervision. Specifically, Slot Attention learns a set of slots that compete with each other through cross attention mechanism to interpret a given image. These slots represent a high level description of objects in the image. To enable slot attention to predict object locations, instead of decoding slots into pixel components, we decode them into individual object coordinates. We supervise the decoded object coordinates outputs with ground truth coordinates by minimizing the MSE loss, where the coordinate matching is achieved with Hungarian Algorithm [64]. To enable this supervised version of Slot Attention to be able to infer object coordinates from out-of-distribution images with object number reaching 8, we set the slot number to be 8. Since the number of ground truth coordinates in training data is K, where $3 < K < 5$, we pad additional $5 - K$ coordinates with values $(1, 1)$ representing empty coordinates (no object) to match the dimension of model output.

Generative Classifier for Object Discovery. Generative Classifier [20] is originally proposed to solve single-label classification problems by using diffusion models, where they try to find the categorical class that minimize denoising error. To infer

object coordinates of continuous values, we adapt Generative Classifier by training a generative model that takes multiple object coordinates as conditioning. During inference, we can inverse the generative model and solve an optimization problems to find a set of object coordinates that best describe the image. For a fair comparison, we train this model following our proposed model. The only difference is that they train a single denoising network taking all coordinates as conditioning, while we train a set of denoising networks each taking an individual object coordinates as conditioning for compositional modeling. During inference, they can follow our proposed inference procedure in Algorithm 5.

ResNet-50 for Facial Attribute Prediction. ResNet-50 [55] has been widely used to solve classification problems. It is straightforward to apply ResNet-50 to solve the facial attribute prediction task. We append a linear layer with input dimension 2048 and output dimension 3 on top of ResNet-50, followed by a sigmoid layer that probability values between 0 and 1. We supervise the model outputs with ground truth facial attribute labels by minimizing the BCE loss. During inference, ResNet-50 choose class label with high probability as classification results.

Generative Classifier for Facial Attribute Prediction. Generative Classifier [20] originally can only solve single-label classification tasks. To enable Generative Classifier to perform multi-label classification, we train a diffusion model taking all three facial attributes as conditioning. During inference, we can enumerate through all possible facial attribute combinations (e.g., combination 1: "black hair, eyeglasses, smiling", combination 2: "not black hair, eyeglasses, smiling", etc.) and evaluate denoising errors. The one combination with smallest denoising error is selected as multi-label classification results. Again, how Generative Classifier in this case differs from our model lies in the lack of compositional modeling.

Generative Classifier Variant for Facial Attribute Prediction. For Generative Classifier Variant, the training procedure is the same as Generative Classifier, but the inference procedure is different. In stead of enumerating attribute combinations, we can evaluate these attributes separately. Specifically, for attribute "black hair", we can evaluate the denoising error of "black hair" and "not black hair" conditioning and determine one of them with smaller denoising error as classification results. We then follow the same procedure to classify other attributes. This inference approach is desired to avoid unaffordable computation complexity when the number of labels is very large.

Diffusion Classifier for Multi-Object Perception. Diffusion Classifier [19] is originally propose to solve zero-shot single-label classification problems by using pretrained text-to-image generative models without requiring any training. To adapt Diffusion Classifier for multi-label classification, we can feed Diffusion Classifier a prompt that describes a combination of multiple concepts as text conditioning and evaluate denoising error. For example, to determine if an image contains cat and dog in our task, Diffusion Classifier can evaluate the denoising error of the following prompts: "a photo of a cat and a dog", "a photo of a cat and a rabbit" and "a photo of a dog and a rabbit". The prompts with smallest denoising error is selected as the classification results.

Diffusion Classifier Variant for Multi-Object Perception. Diffusion Classifier Variant

differs from Diffusion Classifier by evaluating each prompts separately. Given an image containing two animals from a finite set {dog, cat, rabbit}, Diffusion Classifier evaluates the denoising error of following prompts: “a photo of a dog”, “a photo of a cat” and “a photo of a rabbit”, and then choose the two prompts with smallest denoising error as multi-label classification results.

REFERENCES

[1] Y. Wang, J. Dauwels, and Y. Du. "Compositional Scene Understanding through Inverse Generative Modeling". In: *International Conference on Machine Learning*. PMLR. 2025, pp. 62784–62803.

[2] I. Biederman. "Recognition-by-components: a theory of human image understanding." In: *Psychological review* 94.2 (1987), p. 115.

[3] K. Greff, S. Van Steenkiste, and J. Schmidhuber. "On the binding problem in artificial neural networks". In: *arXiv preprint arXiv:2012.05208* (2020).

[4] J. A. Fodor and E. Lepore. *The compositionality papers*. Oxford University Press, 2002.

[5] R. N. Shepard and J. Metzler. "Mental rotation of three-dimensional objects". In: *Science* 171.3972 (1971), pp. 701–703.

[6] N. Chomsky. *Aspects of the Theory of Syntax*. Cambridge: The MIT Press, 1965. URL: http://www.amazon.com/Aspects-Theory-Syntax-Noam-Chomsky/dp/0262530074.

[7] J. A. Fodor and Z. W. Pylyshyn. "Connectionism and cognitive architecture: A critical analysis". In: *Cognition* 28.1-2 (1988), pp. 3–71.

[8] Y. Bengio. "Towards Compositional Understanding of the World by Agent-Based Deep Learning". In: *NeurIPS'2019 Workshop on Context and Compositionality in Biological and Artificial Neural Networks*. 2019.

[9] V. N. Vapnik, V. Vapnik, *et al.* "Statistical learning theory". In: (1998).

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25 (2012).

[11] J. Redmon. "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[12] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness". In: *arXiv preprint arXiv:1811.12231* (2018).

[13] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. "Do imagenet classifiers generalize to imagenet?" In: *International conference on machine learning*. PMLR. 2019, pp. 5389–5400.

[14] R. Taori, A. Dave, V. Shankar, N. Carlini, B. Recht, and L. Schmidt. "Measuring robustness to natural distribution shifts in image classification". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 18583–18599.

[15] D. Hendrycks and K. Gimpel. “A baseline for detecting misclassified and out-of-distribution examples in neural networks”. In: *arXiv preprint arXiv:1610.02136* (2016).

[16] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. “Shortcut learning in deep neural networks”. In: *Nature Machine Intelligence* 2.11 (2020), pp. 665–673.

[17] A. Ng and M. Jordan. “On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes”. In: *Advances in neural information processing systems* 14 (2001).

[18] G. E. Hinton. “To recognize shapes, first learn to generate images”. In: *Progress in brain research* 165 (2007), pp. 535–547.

[19] A. C. Li, M. Prabhudesai, S. Duggal, E. Brown, and D. Pathak. “Your diffusion model is secretly a zero-shot classifier”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2206–2217.

[20] A. C. Li, A. Kumar, and D. Pathak. “Generative Classifiers Avoid Shortcut Solutions”. In: *ICML 2024 Workshop on Structured Probabilistic Inference {\&} Generative Modeling*. 2024.

[21] K. Clark and P. Jaini. “Text-to-image diffusion models are zero shot classifiers”. In: *Advances in Neural Information Processing Systems* 36 (2024).

[22] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. “Deep unsupervised learning using nonequilibrium thermodynamics”. In: *International Conference on Machine Learning*. PMLR. 2015, pp. 2256–2265.

[23] J. Ho, A. Jain, and P. Abbeel. “Denoising diffusion probabilistic models”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6840–6851.

[24] Y. Du and L. Kaelbling. “Compositional generative modeling: A single model is not all you need”. In: *arXiv preprint arXiv:2402.01103* (2024).

[25] P. Jaini, K. Clark, and R. Geirhos. “Intriguing properties of generative classifiers”. In: *arXiv preprint arXiv:2309.16779* (2023).

[26] D. Mahajan, M. Pezeshki, I. Mitliagkas, K. Ahuja, and P. Vincent. “Compositional Risk Minimization”. In: *arXiv preprint arXiv:2410.06303* (2024).

[27] H. Chen, Y. Dong, S. Shao, Z. Hao, X. Yang, H. Su, and J. Zhu. “Your diffusion model is secretly a certifiably robust classifier”. In: *arXiv preprint arXiv:2402.02316* (2024).

[28] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or. “An image is worth one word: Personalizing text-to-image generation using textual inversion”. In: *arXiv preprint arXiv:2208.01618* (2022).

[29] R. Gal, M. Arar, Y. Atzmon, A. H. Bermano, G. Chechik, and D. Cohen-Or. “Encoder-based domain tuning for fast personalization of text-to-image models”. In: *ACM Transactions on Graphics (TOG)* 42.4 (2023), pp. 1–13.

[30] O. Avrahami, K. Aberman, O. Fried, D. Cohen-Or, and D. Lischinski. “Break-a-scene: Extracting multiple concepts from a single image”. In: *SIGGRAPH Asia 2023 Conference Papers*. 2023, pp. 1–12.

[31] T. Amit, T. Shaharbany, E. Nachmani, and L. Wolf. “Segdiff: Image segmentation with diffusion probabilistic models”. In: *arXiv preprint arXiv:2112.00390* (2021).

[32] E. A. Brempong, S. Kornblith, T. Chen, N. Parmar, M. Minderer, and M. Norouzi. “Denoising pretraining for semantic segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 4175–4186.

[33] W. Zhao, Y. Rao, Z. Liu, B. Liu, J. Zhou, and J. Lu. “Unleashing text-to-image diffusion models for visual perception”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 5729–5739.

[34] X. Wang, S. Li, K. Kallidromitis, Y. Kato, K. Kozuka, and T. Darrell. “Hierarchical open-vocabulary universal image segmentation”. In: *Advances in Neural Information Processing Systems* 36 (2024).

[35] Y. Du and I. Mordatch. “Implicit generation and generalization in energy-based models”. In: *arXiv preprint arXiv:1903.08689* (2019).

[36] W. Cho, H. Ravi, M. Harikumar, V. Khuc, K. K. Singh, J. Lu, D. I. Inouye, and A. Kale. *Towards Enhanced Controllability of Diffusion Models*. 2023. DOI: 10.48550/ARXIV.2302.14368. URL: https://arxiv.org/abs/2302.14368.

[37] C. Shi, H. Ni, K. Li, S. Han, M. Liang, G. Mishne, and M. R. Min. “Compositional image generation and manipulation with latent diffusion models”. In: (2023).

[38] K. Sohn, A. Shaw, Y. Hao, H. Zhang, L. Polania, H. Chang, L. Jiang, and I. Essa. “Learning disentangled prompts for compositional image synthesis”. In: *arXiv preprint arXiv:2306.00763* (2023).

[39] Y. Du, S. Li, and I. Mordatch. “Compositional Visual Generation with Energy Based Models”. In: *Advances in Neural Information Processing Systems*. 2020.

[40] Y. Du, S. Li, Y. Sharma, B. J. Tenenbaum, and I. Mordatch. “Unsupervised Learning of Compositional Energy Concepts”. In: *Advances in Neural Information Processing Systems*. 2021.

[41] Y. Du, C. Durkan, R. Strudel, J. B. Tenenbaum, S. Dieleman, R. Fergus, J. Sohl-Dickstein, A. Doucet, and W. Grathwohl. “Reduce, Reuse, Recycle: Compositional Generation with Energy-Based Diffusion Models and MCMC”. In: *arXiv preprint arXiv:2302.11552* (2023).

[42] W. Nie, A. Vahdat, and A. Anandkumar. “Controllable and compositional generation with latent-space energy-based models”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 13497–13510.

[43] W. Feng, X. He, T.-J. Fu, V. Jampani, A. Akula, P. Narayana, S. Basu, X. E. Wang, and W. Y. Wang. “Training-Free Structured Diffusion Guidance for Compositional Text-to-Image Synthesis”. In: *arXiv preprint arXiv:2212.05032* (2022).

[44] S. Li, Y. Du, J. B. Tenenbaum, A. Torralba, and I. Mordatch. “Composing ensembles of pre-trained models via iterative consensus”. In: *arXiv preprint arXiv:2210.11522* (2022).

[45] N. Liu, S. Li, Y. Du, J. Tenenbaum, and A. Torralba. “Learning to compose visual relations”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 23166–23178.

[46] N. Liu, S. Li, Y. Du, A. Torralba, and J. B. Tenenbaum. “Compositional Visual Generation with Composable Diffusion Models”. In: *arXiv preprint arXiv:2206.01714* (2022).

[47] L. Huang, D. Chen, Y. Liu, Y. Shen, D. Zhao, and J. Zhou. “Composer: Creative and Controllable Image Synthesis with Composable Conditions”. In: *arXiv preprint arXiv:2302.09778* (2023).

[48] Y. Cong, M. R. Min, L. E. Li, B. Rosenhahn, and M. Y. Yang. “Attribute-Centric Compositional Text-to-Image Generation”. In: *arXiv preprint arXiv:2301.01413* (2023).

[49] Y. Wang, L. Liu, and J. Dauwels. “Slot-vae: Object-centric scene generation with slot attention”. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 36020–36035.

[50] J. Su*, N. Liu*, Y. Wang*, J. B. Tenenbaum, and Y. Du. “Compositional Image Decomposition with Diffusion Models”. In: *International Conference on Machine Learning*. PMLR. 2024, 46823–46842 (* indicates equal contribution).

[51] S. Zhou, Y. Du, J. Chen, Y. Li, D.-Y. Yeung, and C. Gan. “RoboDreamer: Learning Compositional World Models for Robot Imagination”. In: *arXiv preprint arXiv:2404.12377* (2024).

[52] A. Netanyahu, Y. Du, A. Bronars, J. Pari, J. Tenenbaum, T. Shu, and P. Agrawal. “Few-Shot Task Learning through Inverse Generative Modeling”. In: *arXiv preprint arXiv:2411.04987* (2024).

[53] N. Liu, Y. Du, S. Li, J. B. Tenenbaum, and A. Torralba. “Unsupervised compositional concepts discovery with text-to-image generative models”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2085–2095.

[54] J. Li, D. Li, S. Savarese, and S. Hoi. “Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models”. In: *International conference on machine learning*. PMLR. 2023, pp. 19730–19742.

[55] K. He, X. Zhang, S. Ren, and J. Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[56] F. Locatello, D. Weissenborn, T. Unterthiner, A. Mahendran, G. Heigold, J. Uszkoreit, A. Dosovitskiy, and T. Kipf. *Object-Centric Learning with Slot Attention*. 2020. arXiv: 2006.15055 [cs.LG].

[57] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick. "Clevr: A diagnostic dataset for compositional language and elementary visual reasoning". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2901–2910.

[58] M. Seitzer, M. Horn, A. Zadaianchuk, D. Zietlow, T. Xiao, C.-J. Simon-Gabriel, T. He, Z. Zhang, B. Schölkopf, T. Brox, *et al.* "Bridging the gap to real-world object-centric learning". In: *arXiv preprint arXiv:2209.14860* (2022).

[59] Z. Liu, P. Luo, X. Wang, and X. Tang. "Deep learning face attributes in the wild". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 3730–3738.

[60] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. arXiv: 2112.10752 [cs.CV].

[61] S.-i. Amari. "Backpropagation and stochastic gradient descent method". In: *Neurocomputing* 5.4-5 (1993), pp. 185–196.

[62] L. Karazija, I. Laina, and C. Rupprecht. "Clevrtex: A texture-rich benchmark for unsupervised multi-object segmentation". In: *arXiv preprint arXiv:2111.10265* (2021).

[63] O. Ronneberger, P. Fischer, and T. Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer. 2015, pp. 234–241.

[64] H. W. Kuhn. "The Hungarian method for the assignment problem". In: *Naval research logistics quarterly* 2.1-2 (1955), pp. 83–97.

6

Conclusions, Limitations and Future Directions

6.1. Conclusions

This thesis has explored how compositional generative modeling enables generalization, controllability, and interpretability in several vision tasks. Across the three thrusts including scene generation, scene decomposition, and scene understanding, we consistently found that modeling images as structured compositions of factors of variation provides powerful advantages.

In scene generation, we showed that introducing an object-centric structure into generative models allows manipulation of individual components and yields higher-quality images that generalize more faithfully within the training distribution. In scene decomposition, we demonstrated how representing images as compositions of local and global factors, learned through diffusion-based energy landscapes, enables the unsupervised discovery of reusable visual concepts and supports strong out-of-distribution generalization. Finally, in scene understanding, we formulated scene property inference as a compositional inverse generative modeling problem, showing that this perspective enables object discovery, multi-label classification, and reasoning about scenes more complex than those encountered during training.

Overall, these findings highlight that compositionality is a key inductive bias for building generative models that are not only expressive, but also generalizable and interpretable. By structuring models around objects, factors, and relations, we move toward generative systems that can scale beyond the mimicry of training data to robust reasoning about new and complex scenes.

6.2. Limitations

Despite these advances, our approaches face several important challenges. One limitation lies in the assumption of independence between object concepts when composing diffusion models. Although this assumption simplifies modeling and inference, it does not fully reflect real-world scenarios, where objects interact, coexist, or exhibit relational

dependencies. These interactions, such as physical constraints, semantic relationships, or causal influences, are often essential for accurate reasoning and realistic generation. Extending our models to capture such dependencies, for example, through relational structures, graph-based priors, or joint generative mechanisms, represents a natural and promising direction for future work.

Another challenge concerns scalability in using compositional diffusion models for inference over multiple discrete concepts. In our current framework, inferring object assignments or semantic compositions often requires enumerating possible configurations, an approach that becomes computationally prohibitive as the number of objects, attributes, or categories increases. We have explored a continuous relaxation of discrete variables and discussed heuristic search strategies and parallelization to mitigate this problem; however, these methods only partially alleviate the computational burden. Achieving efficient and scalable inference remains an open and critical challenge.

More broadly, addressing these limitations is essential for moving from compositional modeling of isolated objects toward richer, more holistic scene-level understanding, where objects not only coexist but interact in structured and meaningful ways. Ultimately, tackling these challenges will enable compositional generative models to more closely approximate the relational, context-aware reasoning capabilities found in human cognition.

6.3. FUTURE DIRECTIONS

Several promising avenues remain open for future exploration. One natural extension involves applying compositional generative modeling to dynamic and interactive scenes. Such a direction would enable consistent discovery, tracking, and transformation of objects over time, while also capturing temporal dependencies, causal interactions, and agent behaviors. Modeling not just static structure but also how scenes evolve, such as how objects move, interact, and influence one another, would significantly expand the applicability of compositional models to complex video generation, decomposition, and understanding tasks.

Another exciting direction is to extend compositional representations beyond individual objects to also include their relations. Representing and inferring interactions, such as spatial constraints, semantic relationships, physical dependencies, or even causal influences, would enable models to generate or reason with richer structural descriptions, such as full scene graphs. This would support deeper forms of understanding, including common sense reasoning, relational inference, and structured manipulation of scenes.

These directions point toward a broader vision: generative models that not only learn reusable object-level components, but also their interactions and governing principles, and that can flexibly compose these elements to reason about, imagine, and create novel worlds. Moving toward such models would bring us closer to artificial systems that exhibit compositional reasoning capabilities analogous to human imagination and understanding.

Beyond visual tasks, compositional generative modeling also offers significant potential in other domains. For example, numerous large language models (LLMs) have been trained by different institutions, each encapsulating unique expertise derived from mas-

sive and specialized datasets. Instead of fine-tuning a single model for every downstream task, a promising direction is to compositionally combine multiple expert LLMs to leverage their complementary strengths. Although initial attempts have explored multi-agent LLM debate framework, where models exchange responses and vote to obtain a final answer, more principled approaches are needed to compose their underlying distributions. Such distribution-level composition has the potential to yield more robust, interpretable, and generalizable systems.

Another domain that could greatly benefit from compositional generative modeling is robotic manipulation, where multimodal data, such as vision, language, and action trajectories, must be integrated for reliable and generalizable decision-making. In robotics, collecting training data that captures every possible configuration of environments, objects, and tasks is infeasible. Consequently, monolithic models trained on narrow datasets often struggle to generalize to new combinations or unseen situations. Instead, compositional generative models that learn transferable components, such as object concepts, physical properties, and implicit task constraints, could enable robots to generalize to novel environments by flexibly recombining known building blocks.

We believe that compositional generative models can further benefit many other fields, including scientific discovery, where complex phenomena are often governed by underlying compositional structures. By learning representations based on fundamental building blocks, such as particles, genes, molecules, or physical laws, these models could help uncover latent structures, generate novel hypotheses, and simulate hypothetical scenarios beyond existing datasets. For instance, in chemistry and materials science, they could enable the generation of novel compounds through the recombination of molecular substructures. In biology, compositional models could help infer hierarchical relationships among genes, proteins, or cellular processes, facilitating the discovery of emergent functions. Ultimately, by learning and composing meaningful scientific primitives, compositional generative models could augment human reasoning and accelerate scientific innovation.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincerest gratitude to my promoters, Justin Dauwels and Geert Leus, who have been incredibly supportive and encouraging throughout my PhD journey. Justin inspired me to look beyond the hot research trends of the moment and explore those ideas vital for the future development of AI, such as modularity and compositionality. Along the way, Justin provided me not only invaluable research advice but also the trust and encouragement that allowed me to try diverse ideas, pursue industry internships, and build outside collaborations. I'm equally grateful to Geert, who first connected me with Justin and made the start of this PhD journey possible. In every meeting with Geert, he provided insightful perspectives that pushed my thinking forward. His advice on research definition and communication is invaluable to my development as a researcher.

I am also grateful to my collaborators. I enjoy every discussion with Letao Liu. Although we work on different research topics, I learned a lot from Letao, who was always willing to share his own research and life experiences, including all the ups and downs, encouraging me to push forward. I also had the wonderful opportunity to work with Yilun Du. Despite our many meetings spanning a six-hour time difference, our collaboration was incredibly smooth. From Yilun, I learned a great deal about research motivation, experimentation, writing, and presentation. Furthermore, it was also a great pleasure to work on image decomposition with Jocelin Su and Nan Liu, to have insightful discussions about object-centric learning with Jindong Jiang, and to discuss the application of generative models to smart meter data with Nan Lin. My thanks also go to colleagues at Qualcomm AI Research during my internship.

I am tremendously grateful to the members of my thesis committee, Marcel Reinders, Bert de Vries, Jan van Gemert, Yiyu Chen, and Alle-Jan van der Veen, for their time and dedication in reading, evaluating, and providing insightful comments on this work. In particular, I would like to thank Jan van Gemert, who served as a committee member not only for this thesis but also during my go/no-go meeting. Beyond his formal role, our additional meetings to discuss my research were truly inspiring; I am deeply impressed by his infectious research enthusiasm and rigor.

I am deeply grateful to the exceptionally professional and kind SPS colleagues. I would like to thank our group lead, Alle-Jan, who was always supportive whenever I sought out summer or winter school opportunities. I am also deeply appreciative of Laura, who put great effort into helping me navigate the complicated administrative procedures for my internship leave. Talking with Raj, Richard, Geethu, and Gerard was always a pleasure for their genuine kindness and positive energy.

This PhD journey would not have been as wonderful as it was without my awesome lab mates. The very first group of lab mates I met in the office were Sofia, Ellen and Seline, whose kindness made a lasting impression on me; we started the PhD program at the

same time and shared a lot of our progress along the way. I have had many wonderful, relaxing hours on the tennis court with Anu and Alberto—practicing, playing games, and discussing everything from tennis techniques to life. I also greatly enjoyed the engaging lunch and office conversations with Yanbin, Ids, Metin, Shuoyan, Peiyuan, Zhonggang, Ruben, Sinian, Giovanni, Aybuke, Didem, Chen, Jordi, Cristian, Carlo, Ankush, Miao, Hannie, Costas, Yongsheng, Bishwadeep, Pascal, Rupam, Shao-Hsuan, and Ali. Overall, I want to thank all my lab mates for the coffee breaks, group gatherings, and the stimulating office discussions.

I am very grateful to Changheng and Li. I have no siblings, but you both made me feel as though I do. Our daily conversations and weekly dinners made the rigors of PhD life not only bearable but truly vibrant. When my daughter was born, you waited and slept in the hospital just to be the first ones to see her; I believe that is why she loves you both so much. My thanks also go to Huaizhou for your companionship and support since the start of my time in the Netherlands. Thank you for the delicious meals, the inspiring conversations, and, of course, the countless hours spent on the tennis court. In talking with you, I always learn something interesting and insightful. I also want to thank my other tennis friends Zilong and Jinke.

Finally, I would like to thank my family, my greatest source of strength. I am infinitely grateful to my wife, Yansu Wang. From China to the United States and the Netherlands, you have always been there to support and believe in me; in fact, I think you have always trusted me even more than I trust myself when things are challenging. If there were a 'Best Wife' event in the Olympics, I am certain you would receive the highest score and the gold medal. My daughter, Ruoke Wang, is the greatest gift I was given during this journey, and she is the cutest in the world from head to toe. I am profoundly grateful to my parents, Yu Wang and Qinglian Hu. Thank you for your unwavering support, which has brought me to where I am today. You have always given me the very best of yourselves without reservation; for that, I could not be more grateful. I am also deeply appreciative of the value you place on education and lifelong learning, which has been the foundation of my own journey.

Yanbo Wang
Delft, March 2026