Compositional Image Decomposition with Diffusion Models

Jocelin Su [1 *] **Nan Liu** [2 *] **Yanbo Wang** [3 *] **Joshua B. Tenenbaum** [1] **Yilun Du** [1]

Abstract

Given an image of a natural scene, we are able to quickly decompose it into a set of components such as objects, lighting, shadows, and foreground. We can then envision a scene where we combine certain components with those from other images, for instance a set of objects from our bedroom and animals from a zoo under the lighting conditions of a forest, even if we have never encountered such a scene before. In this paper, we present a method to decompose an image into such compositional components. Our 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. Code and visualizations are at https://energy-based-model.github.io/decomp-diffusion.

1 Introduction

Humans have the remarkable ability to quickly learn new concepts, such as learning to use a new tool after observing just a few demonstrations (Allen et al., 2020). This skill relies on the ability to combine and reuse previously acquired concepts to accomplish a given task (Lake et al., 2017). 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 (Chomsky, 1965). 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 (Vedantam et al., 2018; Higgins et al., 2018; Singh et al., 2019; Peebles et al., 2020). 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 (Burgess et al., 2019; Locatello et al., 2020b; Monnier et al., 2021; Engelcke et al., 2021a). 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 (Du et al., 2021a) 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 paper, we leverage the close connection between Energy-Based Models (LeCun et al., 2006; Du & Mordatch, 2019) and diffusion models (Sohl-Dickstein et al., 2015; Ho

[*]Equal contribution [1]MIT [2]UIUC [3]TU Delft. Correspondence to: Jocelin Su <jocelin@mit.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

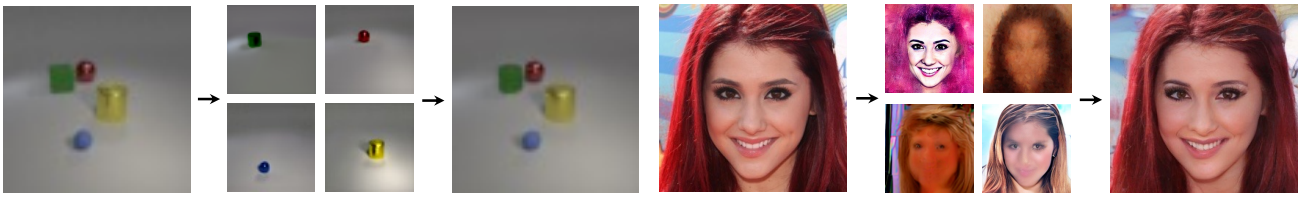

Figure 1: **Image Decomposition with Decomp Diffusion.** Our unsupervised method can decompose an input image into both local factors, such as objects (**Left**), and global factors (**Right**), such as facial features. Additionally, our approach can combine the deduced factors for image reconstruction.

et al., 2020) 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 (Liu et al., 2022; Du et al., 2023), as illustrated in Figure 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. In summary, we contribute the following: 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.

2 Unsupervised Decomposition of Images into Energy Functions

In this section, we introduce background information about COMET (Du et al., 2021a), 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:

$$\underset{\boldsymbol{x}}{\text{argmin}} \sum_k E_\theta(\boldsymbol{x}; \boldsymbol{z}_k). \tag{1}$$

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 $\boldsymbol{x}_i$:

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

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

$$\boldsymbol{x}_i^N = \boldsymbol{x}_i^{N-1} - \gamma \nabla_{\boldsymbol{x}} \sum_k E_\theta(\boldsymbol{x}_i^{N-1}; \text{Enc}_\phi(\boldsymbol{x}_i)[k]), \tag{3}$$

where γ is the step size. Optimizing the training objective in Equation 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.

3 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 3.1. Then in Section 3.2, we describe how we use this connection in Decomp Diffusion to decompose images into a set of composable diffusion models.

3.1 Denoising Networks as Energy Functions

Denoising Diffusion Probabilistic Models (DDPMs) (Sohl-Dickstein et al., 2015; Ho et al., 2020) 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 $\epsilon \sim \mathcal{N}(0, 1)$,

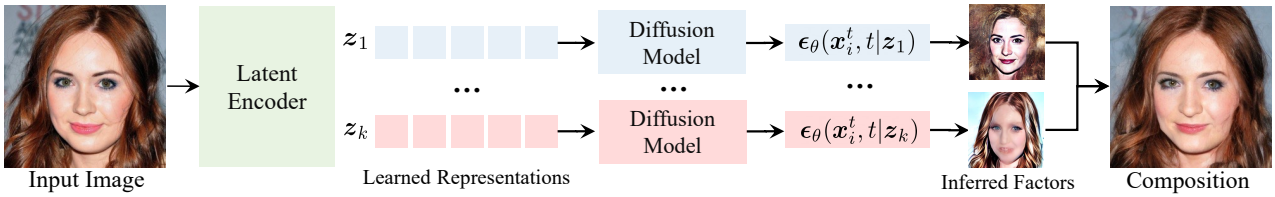

Figure 2: **Compositional Image Decomposition**. We learn to decompose each input image into a set of denoising functions $\{\epsilon_\theta(\boldsymbol{x}_i^t, t, |\boldsymbol{z}_k)\}$ representing K factors, which can be composed to reconstruct the input.

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:

$$\mathcal{L}_{\text{MSE}} = \|\epsilon - \epsilon_\theta(\sqrt{1-\beta_t}\boldsymbol{x}_i + \sqrt{\beta_t}\epsilon, t)\|_2^2. \quad (4)$$

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 (Ho et al., 2020). As discussed in both (Liu et al., 2022; Du et al., 2023), this gradient field also corresponds to the gradient field of an energy function

$$\epsilon_\theta(\boldsymbol{x}^t, t) = \nabla_{\boldsymbol{x}} E_\theta(\boldsymbol{x}) \quad (5)$$

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

$$\boldsymbol{x}^{t-1} = \boldsymbol{x}^t - \gamma\epsilon_\theta(\boldsymbol{x}^t, t) + \xi, \quad \xi \sim \mathcal{N}\big(0, \sigma_t^2 I\big), \quad (6)$$

where σ_t^2 is the variance according to a variance schedule and γ is the step size[1]. This directly corresponds to the noisy energy optimization procedure

$$\boldsymbol{x}^{t-1} = \boldsymbol{x}^t - \gamma\nabla_{\boldsymbol{x}} E_\theta(\boldsymbol{x}^t) + \xi, \quad \xi \sim \mathcal{N}\big(0, \sigma_t^2 I\big). \quad (7)$$

The functional form of Equation 7 is very similar to Equation 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 intermediate latent $\boldsymbol{z}$, then the denoising network directly learns an $\epsilon_\theta(\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{z})$ that estimates gradients of an energy function $\nabla_{\boldsymbol{x}} E_\theta(\boldsymbol{x}, \boldsymbol{z})$. This energy function has minimum

$$\boldsymbol{x}_i = \underset{\boldsymbol{x}}{\operatorname{argmin}}\, E_\theta(\boldsymbol{x}, \boldsymbol{z}), \quad (8)$$

as the highest log-likelihood datapoint will be $\boldsymbol{x}_i$. 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 2.

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

Algorithm 1 Training Algorithm

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

3.2 Decompositional Diffusion Models

In COMET, given an input image $\boldsymbol{x}_i$, we are interested in inferring a set of different latent energy functions $E_\theta(\boldsymbol{x}, \boldsymbol{z}_k)$ such that

$$\boldsymbol{x}_i = \underset{\boldsymbol{x}}{\operatorname{argmin}} \sum_k E_\theta(\boldsymbol{x}, \boldsymbol{z}_k).$$

Using the equivalence between denoising networks and energy function discussed in Section 3.1 to recover the desired set of energy functions, we may simply learn a set of different denoising functions to recover an image $\boldsymbol{x}_i$ using the objective:

$$\mathcal{L}_{\text{MSE}} = \left\|\epsilon - \textstyle\sum_k \epsilon_\theta\left(\sqrt{1-\beta_t}\boldsymbol{x}_i + \sqrt{\beta_t}\epsilon, t, \boldsymbol{z}_k\right)\right\|_2^2, \quad (9)$$

where each individual latent $\boldsymbol{z}_k$ is inferred by a jointly learned neural network encoder $\text{Enc}_\phi(\boldsymbol{x}_i)[k]$. We leverage information bottleneck to encourage components to discover independent portions of $\boldsymbol{x}_i$ by constraining latent representations $\boldsymbol{z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_K\}$ 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 (Ho et al., 2020), the denoising network ϵ_θ may either be trained to directly estimate the starting noise ϵ or the original image $\boldsymbol{x}_i$. These two predictions are functionally identical, as ϵ can be directly obtained by taking a linear combination of noisy image $\boldsymbol{x}^t$ and $\boldsymbol{x}_i$. While standard diffusion training directly

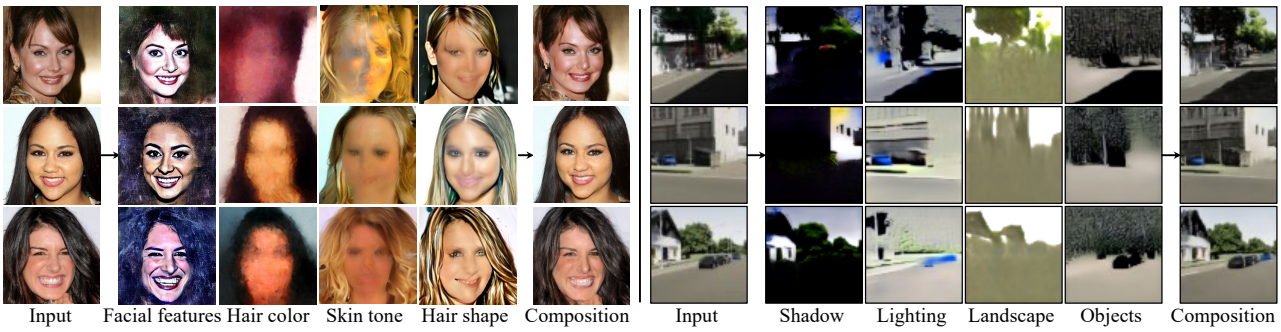

Figure 3: **Global Factor Decomposition.** Our method can enable global factor decomposition and reconstruction on CelebA-HQ (**Left**) and Virtual KITTI 2 (**Right**). *Note that discovered factors are labeled with posited factors.*

Algorithm 2 Image Generation Algorithm

Input: Diffusion steps T, denoising model ϵ_θ, latent vectors $\{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K\}$, step size γ
$\boldsymbol{x}^T \sim \mathcal{N}(0, 1)$
for $t = T, \ldots, 1$ **do**
  ▷ *Sample Gaussian noise*
  $\xi \sim \mathcal{N}(0, 1)$
  ▷ *Compute denoising direction*
  $\epsilon_{\text{pred}} \leftarrow \sum_k \epsilon_\theta(\boldsymbol{x}^t, t, \boldsymbol{z}_k)$
  ▷ *Run noisy gradient descent*
  $\boldsymbol{x}^{t-1} = \frac{1}{\sqrt{1-\beta_t}}(\boldsymbol{x}^t - \gamma\epsilon_{\text{pred}} + \sqrt{\beta_t}\xi)$
end for

predicts ϵ, we find that predicting $\boldsymbol{x}_i$ and then regressing ϵ leads to better performance, as this training objective is more similar to autoencoder training.

Once we have recovered these denoising functions, we may directly use the noisy optimization objective in Equation 7 to sample from compositions of different factors. The full training and sampling algorithm for our approach are shown in Algorithm 1 and Algorithm 2 respectively.

4 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 4.2. We next evaluate decomposition of images into local factors of variation in Section 4.3. We further investigate the ability of decomposed components to recombine across separate trained models in Section 4.4. Finally, we illustrate how our approach can be adapted to pretrained models in Section 4.5. 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.

4.1 Quantitative Metrics

For quantitative evaluation of image quality, we employ Fréchet Inception Distance (FID) (Heusel et al., 2017), Kernel Inception Distance (KID) (Bińkowski et al., 2018), and LPIPS (Zhang et al., 2018) on images reconstructed from CelebA-HQ (Karras et al., 2017), Falcor3D (Nie et al., 2020), Virtual KITTI 2 (Cabon et al., 2020), and CLEVR (Johnson et al., 2017). To evaluate disentanglement, we compute MIG (Chen et al., 2018) and MCC (Hyvärinen & Morioka, 2016) on learned latent representation images on the Falcor3D dataset.

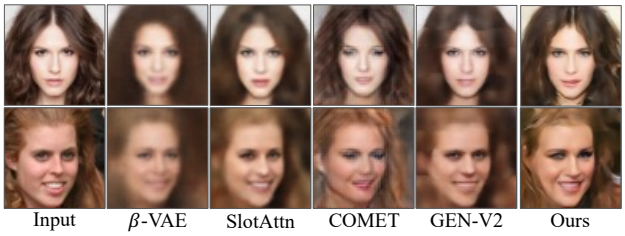

Figure 4: **Reconstruction comparison.** Our method can reconstruct input images with a high fidelity on CelebA-HQ.

4.2 Global Factors

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 3, we show 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 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. 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 3, we show how Decomp Diffusion infers factors such as shadow, lighting, landscape, and

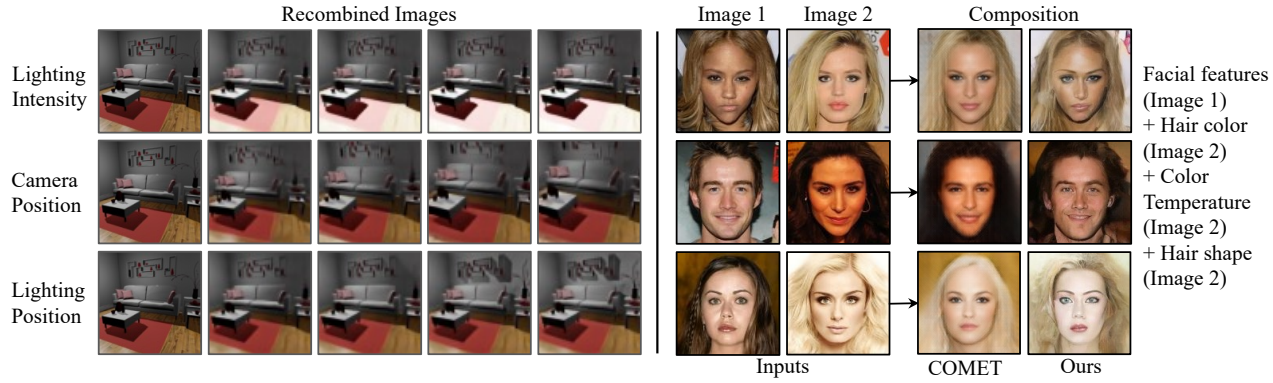

Figure 5: **Global Factor Recombination.** Recombination of inferred factors on Falcor3D and CelebA-HQ datasets. In Falcor3D (**Left**), we show image variations by varying inferred factors such as lighting intensity. In CelebA-HQ (**Right**), we recombine factors from two different inputs to generate novel face combinations.

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 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.

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

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 19.

We also provide qualitative results to illustrate the effect of number of concepts K on CelebA-HQ and Falcor3D in Figure 17 and Figure 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 18.

Recombination. In Figure 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 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 1. We observe that our method outperforms existing methods in terms of FID, KID, and LPIPS across datasets, indicating superior image reconstruction quality.

Finally, we evaluate the disentanglement of the given methods on the Falcor3D dataset. As shown in Table 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

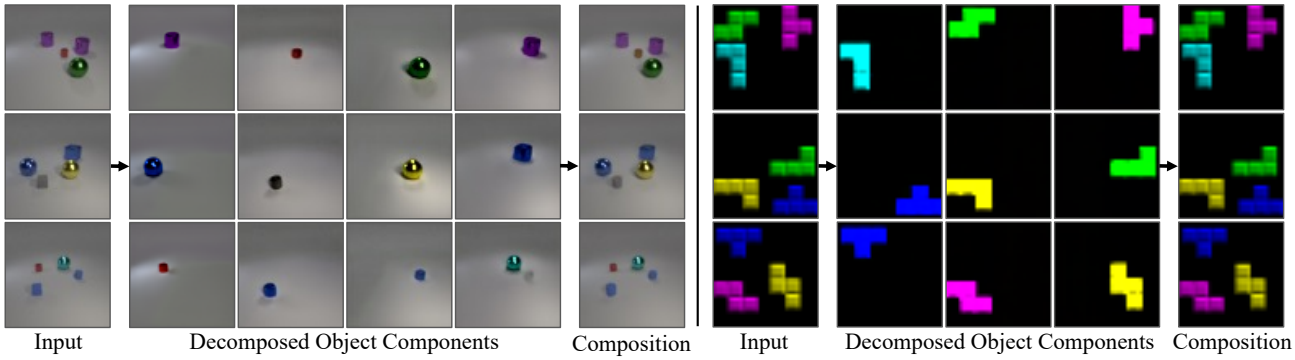

Figure 6: **Local Factor Decomposition.** Illustration of object-level decomposition on CLEVR (**left**) and Tetris (**right**). Our method can extract individual object components that can be reused for image reconstruction.

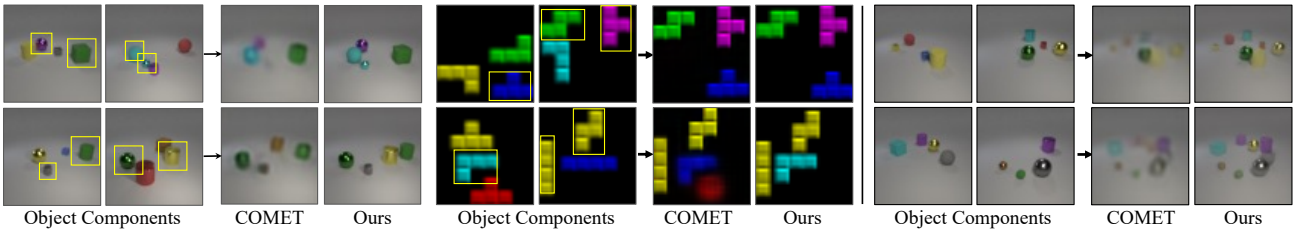

Figure 7: **Local Factor Recombination.** We recombine local factors from 2 images to generate composition of inferred object factors. On both CLEVR and Tetris (**Left**), we recombine inferred object components in the bounding box to generate novel object compositions. On CLEVR (**Right**), we compose all inferred factors to generalize up to 8 objects, though training images only contain 4 objetcs.

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 3. We find that directly predicting the input $\boldsymbol{x}_0$ (3rd and 6th rows) outperforms the ϵ parametrization (1st and 4th row) on both CelebA-HQ and CLEVR datasets in terms of MSE and LPIPS (Zhang et al., 2018). This is due to using a reconstruction-based training procedure, as discussed in Section 3.2. We also compare using a single component to learn reconstruction (2nd and 5th rows) with our method (3rd and 6th rows), which uses multiple components for reconstruction. Our method achieves the best reconstruction quality as measured by MSE and LPIPS.

4.3 Local Factors

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.

Dataset	Multiple Components	Predict $\boldsymbol{x}_0$	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 3: **Ablations.** We analyze the impact of predicting $\boldsymbol{x}_0$ 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.

Decomposition and Reconstructions. We qualitatively evaluate local factor decomposition on object datasets such as CLEVR and Tetris in Figure 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 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 representa-

tions.

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 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.

4.4 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 a greater challenge for extracting common factors. Despite this, we demonstrate our method's success in this regard in Figure 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 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 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 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.

4.5 Decomposition 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 1. 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 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 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.

5 Related Work

Compositional Generation. Existing work on compositional generation study either modifying the underlying generative process to focus on a set of specifications (Feng et al., 2022; Shi et al., 2023; Cong et al., 2023; Huang et al., 2023; Garipov et al., 2023), or composing a set of independent models specifying desired constraints (Du et al., 2020; Liu et al., 2021; 2022; Nie et al., 2021; Du et al., 2023; Wang et al., 2023b). Similar to (Du et al., 2021b), our work aims discover a set of compositional components from an unlabeled dataset of images which may further be integrated with compositional operations from (Du et al., 2023; Liu et al., 2022).

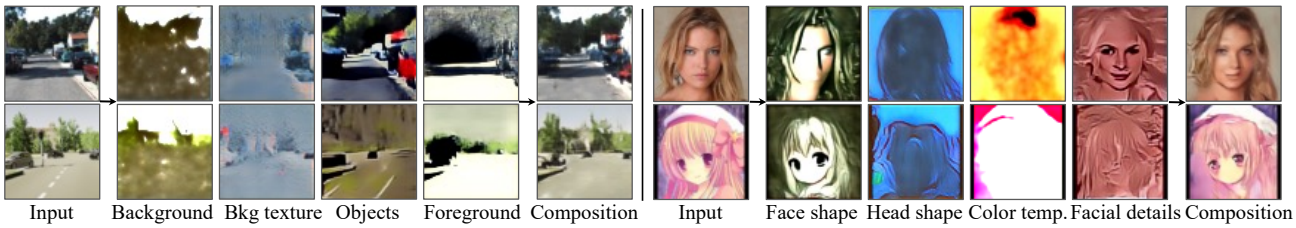

Figure 8: **Multi-modal Dataset Decomposition.** We show our method can capture a set of global factors that are shared between hybrid datasets such as KITTI and Virtual KITTI 2 scenes (**Left**), and CelebA-HQ and Anime faces (**Right**). *Note that discovered factors are labeled with posited factors.*

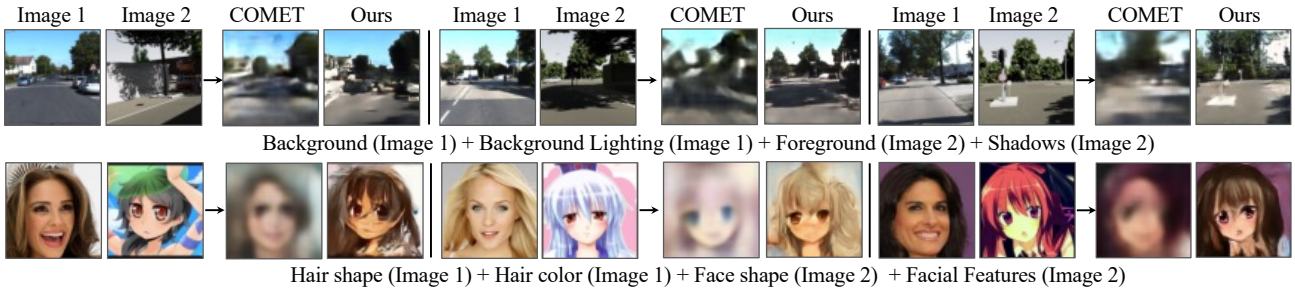

Figure 9: **Multi-modal Dataset Recombination.** Our method exhibits the ability to recombine inferred factors from various hybrid datasets. We can recombine different extracted factors to generate unique compositions of KITTI and Virtual KITTI 2 scenes (**Top**), and compositions of CelebA-HQ and Anime faces (**Bottom**).

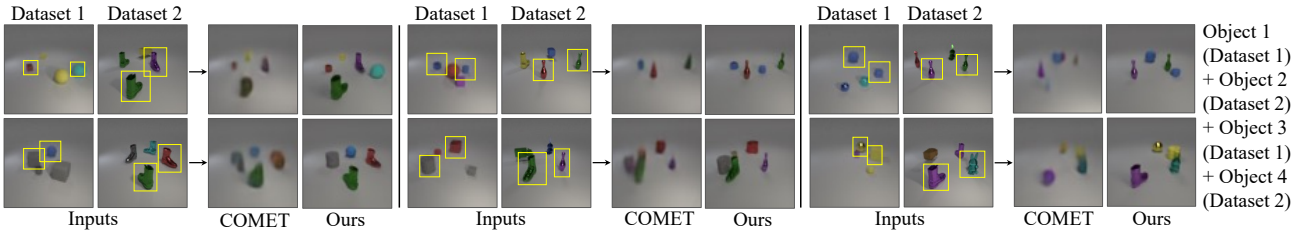

Figure 10: **Cross Dataset Recombination.** We further showcase our method's ability to recombine across datasets using 2 different models that train on CLEVR and CLEVR Toy, respectively. We compose inferred factors as shown in the bounding box from two different modalites to generate unseen compositions.

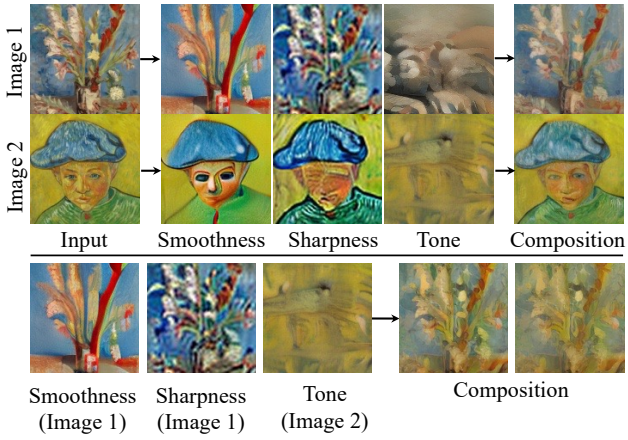

Figure 11: **Art Style Decomposition and Recombination.** Illustration of art style decomposition on a Van Gogh painting dataset. Our method can discover art components that capture different facets of the painting content. The discovered factors can be recombined across images to generate novel images.

Unsupervised Decomposition. Unsupervised decomposition focuses on discovering a global latent space which best describes the input space (Higgins et al., 2017; Burgess et al., 2018; Locatello et al., 2020a; Klindt et al., 2021; Peebles et al., 2020; Singh et al., 2019; Preechakul et al., 2022). 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 (Du et al., 2021a), 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 (Burgess et al., 2019; Greff et al., 2019; Locatello et al., 2020b; Engelcke et al., 2021a; Kipf et al., 2022; Seitzer et al., 2022; Wang et al., 2023a), 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 (Jiang et al., 2023; Wu et al., 2023), 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 (Lee et al., 2023; Chefer et al., 2023; Avrahami et al., 2023a; Li et al., 2023; Avrahami et al., 2023b; Kumari et al., 2023; Wei et al., 2023; Shah et al., 2023), 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 (Liu et al., 2023a; Yi et al., 2023; Song et al., 2023) or text captions (Xu et al., 2022), while our decomposition approach is completely unsupervised. The most relevant work to ours, (Liu et al., 2023b) 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.

6 Conclusion

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 C, 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.

Conclusion. In this work, 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.

Acknowledgements

We acknowledge support from NSF grant 2214177; from AFOSR grant FA9550-22-1-0249; from ONR MURI grant N00014-22-1-2740; and from ARO grant W911NF-23-1-0034. Yilun Du is supported by a NSF Graduate Fellowship.

Impact Statement

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.

References

Allen, K. R., Smith, K. A., and Tenenbaum, J. B. Rapid trial-and-error learning with simulation supports flexible tool use and physical reasoning. *Proceedings of the National Academy of Sciences*, 117(47):29302–29310, 2020. ISSN 0027-8424. doi: 10.1073/pnas.1912341117. URL https://www.pnas.org/content/117/47/29302. 1

Avrahami, O., Aberman, K., Fried, O., Cohen-Or, D., and Lischinski, D. Break-a-scene: Extracting multiple concepts from a single image. In *SIGGRAPH Asia 2023 Conference Papers*, pp. 1–12, 2023a. 9

Avrahami, O., Hertz, A., Vinker, Y., Arar, M., Fruchter, S., Fried, O., Cohen-Or, D., and Lischinski, D. The chosen one: Consistent characters in text-to-image diffusion models. *arXiv preprint arXiv:2311.10093*, 2023b. 9

Bińkowski, M., Sutherland, D. J., Arbel, M., and Gretton, A. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 4

Branwen, G., Anonymous, and Community, D. Danbooru2019 portraits: A large-scale anime head illustration dataset, 2019. 16

Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., and Lerchner, A. Understanding disentangling in beta-vae. *arXiv preprint arXiv:1804.03599*, 2018. 8

Burgess, C. P., Matthey, L., Watters, N., Kabra, R., Higgins, I., Botvinick, M., and Lerchner, A. Monet:

Unsupervised scene decomposition and representation. *arXiv:1901.11390*, 2019. 1, 8, 17

Cabon, Y., Murray, N., and Humenberger, M. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 4, 16

Chefer, H., Lang, O., Geva, M., Polosukhin, V., Shocher, A., Irani, M., Mosseri, I., and Wolf, L. The hidden language of diffusion models. *arXiv preprint arXiv:2306.00966*, 2023. 9

Chen, T. Q., Li, X., Grosse, R., and Duvenaud, D. Isolating sources of disentanglement in variational autoencoders. *arXiv:1802.04942*, 2018. 4

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *NeurIPS*, 2016. 17

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

Cong, Y., Min, M. R., Li, L. E., Rosenhahn, B., and Yang, M. Y. Attribute-centric compositional text-to-image generation. *arXiv preprint arXiv:2301.01413*, 2023. 7

Du, Y. and Mordatch, I. Implicit generation and generalization in energy-based models. *arXiv preprint arXiv:1903.08689*, 2019. 1

Du, Y., Li, S., and Mordatch, I. Compositional visual generation with energy based models. In *Advances in Neural Information Processing Systems*, 2020. 7

Du, Y., Li, S., Sharma, Y., Tenenbaum, B. J., and Mordatch, I. Unsupervised learning of compositional energy concepts. In *Advances in Neural Information Processing Systems*, 2021a. 1, 2, 8, 17

Du, Y., Smith, K. A., Ullman, T., Tenenbaum, J. B., and Wu, J. Unsupervised discovery of 3d physical objects. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=lf7st0bJIA5. 7

Du, Y., Durkan, C., Strudel, R., Tenenbaum, J. B., Dieleman, S., Fergus, R., Sohl-Dickstein, J., Doucet, A., and Grathwohl, W. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. *arXiv preprint arXiv:2302.11552*, 2023. 2, 3, 7

Engelcke, M., Parker Jones, O., and Posner, I. Genesis-v2: Inferring unordered object representations without iterative refinement. *Advances in Neural Information Processing Systems*, 34:8085–8094, 2021a. 1, 8

Engelcke, M., Parker Jones, O., and Posner, I. GENESIS-V2: Inferring Unordered Object Representations without Iterative Refinement. *arXiv preprint arXiv:2104.09958*, 2021b. 18

Feng, W., He, X., Fu, T.-J., Jampani, V., Akula, A., Narayana, P., Basu, S., Wang, X. E., and Wang, W. Y. Training-free structured diffusion guidance for compositional text-to-image synthesis. *arXiv preprint arXiv:2212.05032*, 2022. 7

Garipov, T., De Peuter, S., Yang, G., Garg, V., Kaski, S., and Jaakkola, T. Compositional sculpting of iterative generative processes. *Advances in neural information processing systems*, 36:12665–12702, 2023. 7

Geiger, A., Lenz, P., and Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 16

Greff, K., Kaufmann, R. L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M., and Lerchner, A. Multi-object representation learning with iterative variational inference. *arXiv preprint arXiv:1903.00450*, 2019. 8, 16

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pp. 6626–6637, 2017. 4

Higgins, I., Matthey, L., Pal, A., Burgess, C. P., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. Beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017. 8, 17

Higgins, I., Sonnerat, N., Matthey, L., Pal, A., Burgess, C. P., Bosnjak, M., Shanahan, M., Botvinick, M., Hassabis, D., and Lerchner, A. Scan: Learning hierarchical compositional visual concepts. *ICLR*, 2018. 1

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 2, 3, 16

Huang, L., Chen, D., Liu, Y., Shen, Y., Zhao, D., and Zhou, J. Composer: Creative and controllable image synthesis with composable conditions. *arXiv preprint arXiv:2302.09778*, 2023. 7

Hyvärinen, A. and Morioka, H. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. In *Advances in Neural Information Processing Systems*, pp. 3765–3773, 2016. 4

Jiang, J., Deng, F., Singh, G., and Ahn, S. Object-centric slot diffusion, 2023. 9

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017. 4

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. In *ICLR*, 2017. 4

Kipf, T., Elsayed, G. F., Mahendran, A., Stone, A., Sabour, S., Heigold, G., Jonschkowski, R., Dosovitskiy, A., and Greff, K. Conditional object-centric learning from video, 2022. 8

Klindt, D. A., Schott, L., Sharma, Y., Ustyuzhaninov, I., Brendel, W., Bethge, M., and Paiton, D. Towards nonlinear disentanglement in natural data with temporal sparse coding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=EbIDjBynYJ8. 8

Kumari, N., Zhang, B., Zhang, R., Shechtman, E., and Zhu, J.-Y. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1931–1941, 2023. 9

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. Building machines that learn and think like people. *Behav. Brain Sci.*, 40, 2017. 1

LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006. 1

Lee, S., Zhang, Y., Wu, S., and Wu, J. Language-informed visual concept learning. In *The Twelfth International Conference on Learning Representations*, 2023. 9

Li, Z., Cao, M., Wang, X., Qi, Z., Cheng, M.-M., and Shan, Y. Photomaker: Customizing realistic human photos via stacked id embedding. *arXiv preprint arXiv:2312.04461*, 2023. 9

Liu, F., Liu, Y., Kong, Y., Xu, K., Zhang, L., Yin, B., Hancke, G. P., and Lau, R. W. H. Referring image segmentation using text supervision. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22067–22077, 2023a. URL https://api.semanticscholar.org/CorpusID:261243179. 9

Liu, N., Li, S., Du, Y., Tenenbaum, J., and Torralba, A. Learning to compose visual relations. *Advances in Neural Information Processing Systems*, 34:23166–23178, 2021. 7

Liu, N., Li, S., Du, Y., Torralba, A., and Tenenbaum, J. B. Compositional visual generation with composable diffusion models. *arXiv preprint arXiv:2206.01714*, 2022. 2, 3, 7

Liu, N., Du, Y., Li, S., Tenenbaum, J. B., and Torralba, A. Unsupervised compositional concepts discovery with text-to-image generative models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2085–2095, 2023b. 9, 18

Locatello, F., Tschannen, M., Bauer, S., Rätsch, G., Schölkopf, B., and Bachem, O. Disentangling factors of variation using few labels, 2020a. 8

Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., and Kipf, T. Object-centric learning with slot attention, 2020b. 1, 8, 18

Monnier, T., Vincent, E., Ponce, J., and Aubry, M. Unsupervised layered image decomposition into object prototypes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8640–8650, 2021. 1

Nie, W., Karras, T., Garg, A., Debnath, S., Patney, A., Patel, A. B., and Anandkumar, A. Semi-supervised stylegan for disentanglement learning. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 7360–7369, 2020. 4

Nie, W., Vahdat, A., and Anandkumar, A. Controllable and compositional generation with latent-space energy-based models. *Advances in Neural Information Processing Systems*, 34, 2021. 7

Peebles, W., Peebles, J., Zhu, J.-Y., Efros, A., and Torralba, A. 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*, pp. 581–597. Springer, 2020. 1, 8

Preechakul, K., Chatthee, N., Wizadwongsa, S., and Suwajanakorn, S. Diffusion autoencoders: Toward a meaningful and decodable representation, 2022. 8

Seitzer, M., Horn, M., Zadaianchuk, A., Zietlow, D., Xiao, T., Simon-Gabriel, C.-J., He, T., Zhang, Z., Schölkopf, B., Brox, T., et al. Bridging the gap to real-world object-centric learning. *arXiv preprint arXiv:2209.14860*, 2022. 8

Shah, V., Ruiz, N., Cole, F., Lu, E., Lazebnik, S., Li, Y., and Jampani, V. Ziplora: Any subject in any style by effectively merging loras. *arXiv preprint arXiv:2311.13600*, 2023. 9

Shi, C., Ni, H., Li, K., Han, S., Liang, M., and Min, M. R. Exploring compositional visual generation with latent classifier guidance. *arXiv preprint arXiv:2304.12536*, 2023. 7

Singh, K. K., Ojha, U., and Lee, Y. J. Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6490–6499, 2019. 1, 8

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015. 1, 2

Song, Y., Zhang, Z., Lin, Z., Cohen, S., Price, B., Zhang, J., Kim, S. Y., and Aliaga, D. Objectstitch: Object compositing with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18310–18319, 2023. 9

Vedantam, R., Fischer, I., Huang, J., and Murphy, K. Generative models of visually grounded imagination. In *ICLR*, 2018. 1

Wang, Y., Liu, L., and Dauwels, J. Slot-vae: Object-centric scene generation with slot attention. In *International Conference on Machine Learning*, pp. 36020–36035. PMLR, 2023a. 8

Wang, Z., Gui, L., Negrea, J., and Veitch, V. Concept algebra for text-controlled vision models. *arXiv preprint arXiv:2302.03693*, 2023b. 7

Wei, Y., Zhang, Y., Ji, Z., Bai, J., Zhang, L., and Zuo, W. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15943–15953, 2023. 9

Wu, Z., Hu, J., Lu, W., Gilitschenski, I., and Garg, A. Slot-diffusion: Object-centric generative modeling with diffusion models. *arXiv preprint arXiv:2305.11281*, 2023. 9

Xu, J., Mello, S. D., Liu, S., Byeon, W., Breuel, T., Kautz, J., and Wang, X. Groupvit: Semantic segmentation emerges from text supervision. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18113–18123, 2022. URL https://api.semanticscholar.org/CorpusID:247026092. 9

Yi, M., Cui, Q., Wu, H., Yang, C., Yoshie, O., and Lu, H. A simple framework for text-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7071–7080, June 2023. 9

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep networks as a perceptual metric. In *CVPR*, 2018. 4, 6

A Overview

In this supplementary material, we present additional qualitative results for various domains in Section B. Next, we describe the model architecture for our approach in Section D. Finally, we include experiment details on training datasets, baselines, training, and inference in Section E.

B Additional Results

We first provide additional results on global factor decomposition and recombination in Section B.1. We then give additional results on object-level decomposition and recombination in Section B.2. Finally, we provide more results that demonstrate cross-dataset generalization in Section B.3.

B.1 Global Factors

Decomposition and Reconstruction. In Figure 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 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 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.

B.2 Local Factors

Decomposition and Reconstruction. We present additional results for local scene decomposition in Figure 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 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.

B.3 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, *i.e.*, a dataset that consists of different types of images. On the left side of Figure 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 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 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.

C 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 17 and Figure 18.

Decomposition Comparisons. We provide qualitative comparisons of decomposed concepts in Figure 19 and Figure 21.

Factor Semantics. To visualize the impact of each decomposed factor, in Figure 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 23. Our method can iteratively incorporate each object represented by the learned

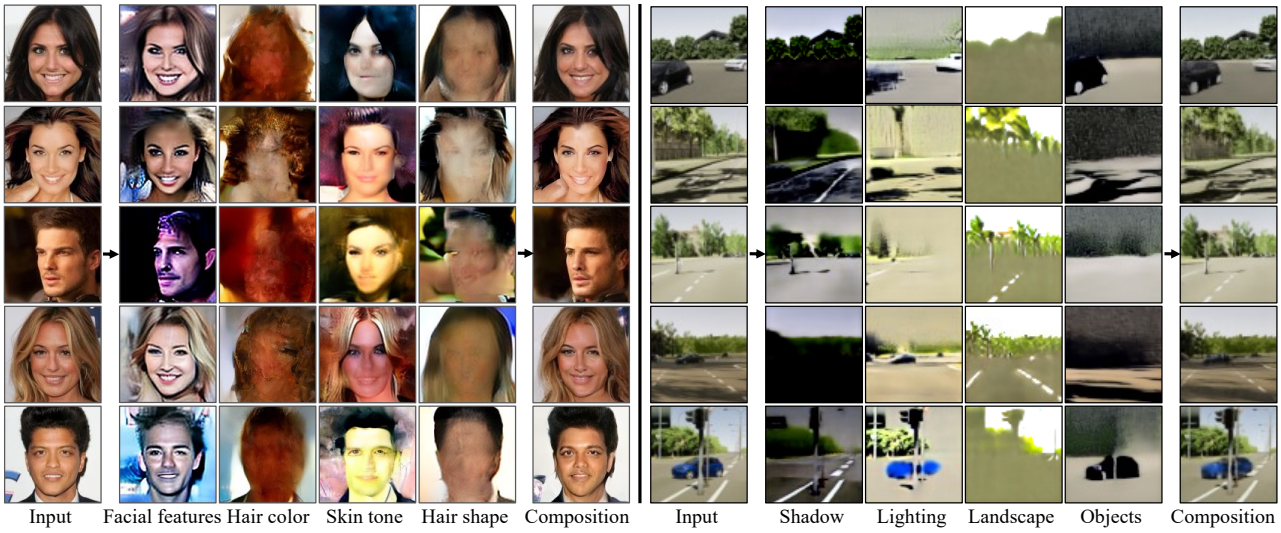

Figure 12: **Global Factor Decomposition.** Global factor decomposition and composition results on CelebA-HQ and Virtual KITTI 2. Note that we name inferred concepts for easier understanding.

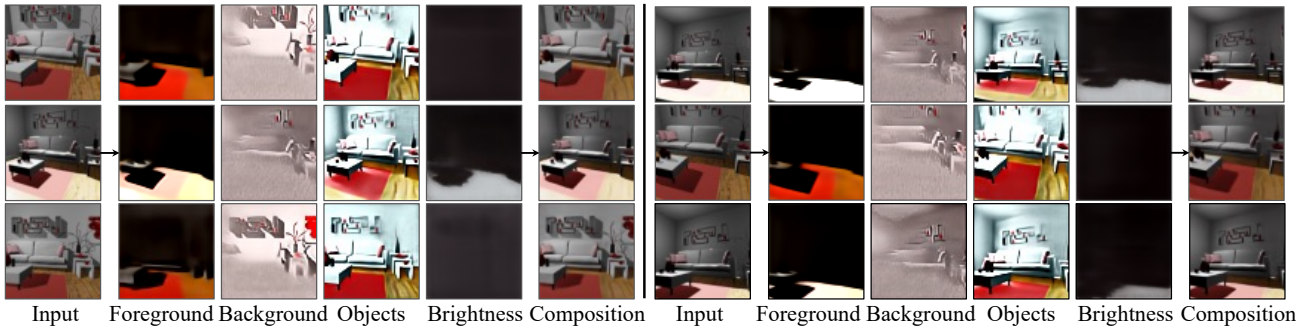

Figure 13: **Global Factor Decomposition.** Global factor decomposition and composition results on Falcor3D. Note that we name inferred concepts for easier understanding.

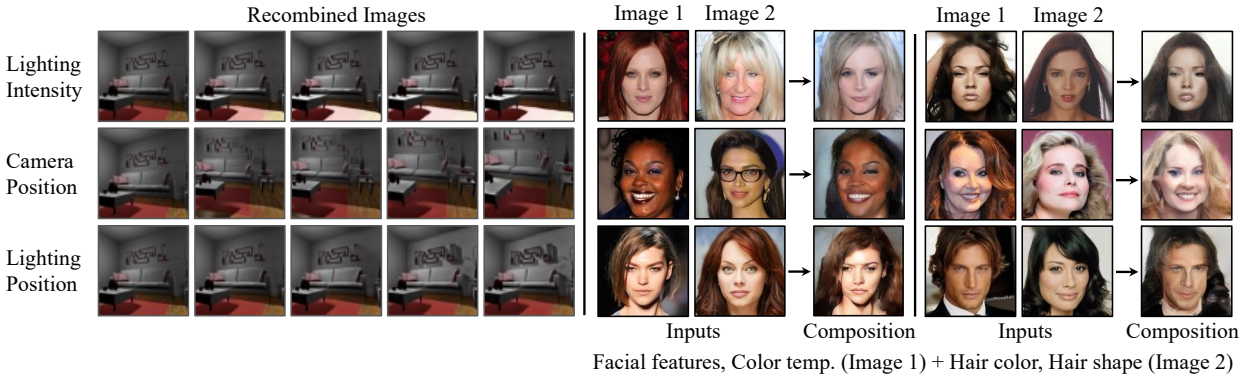

Figure 14: **Global Factor Recombination.** Recombination of inferred factors on Falcor3D and CelebA-HQ datasets. In Falcor3D (**Left**), we show image variations by varying inferred factors such as lighting intensity. In CelebA-HQ (**Right**), we recombine factors from two different inputs to generate novel face combinations.

local factors until it reconstructs the original image's object setup.

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

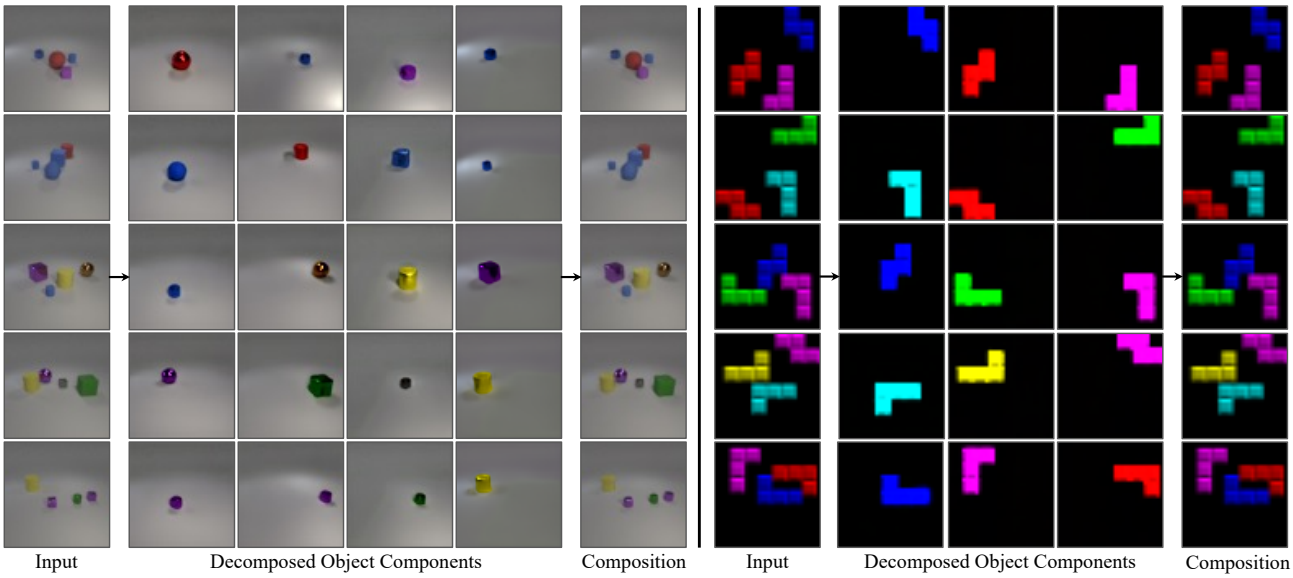

Figure 15: **Local Factor Decomposition.** Object-level decompositions results on CLEVR (**left**) and Tetris (**right**).

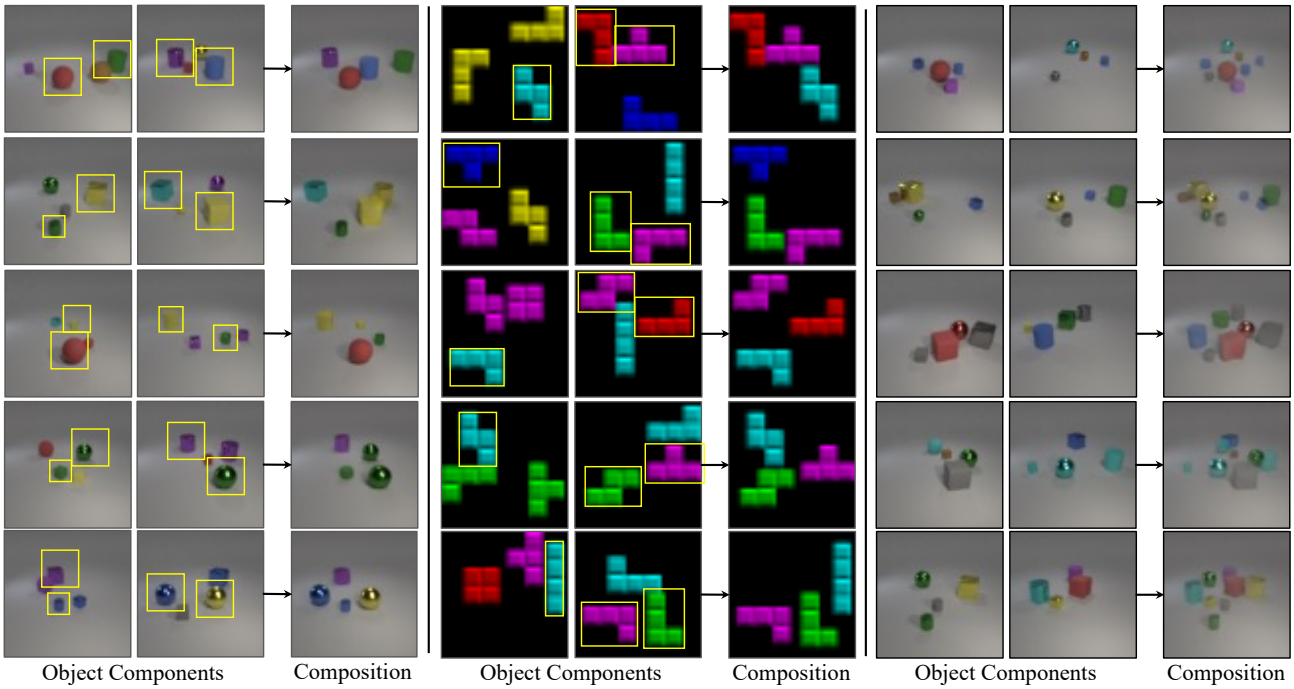

Figure 16: **Local Factor Recombination.** Recombination results using object-level factors from different images.

less significant, indicated by their lower weights. The remaining $K - K'$ components may offer a more suitable fit for the dataset. In Figure 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 25, since the method only optimizes the word embedding in the text encoder 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 26. We can see that just using the pretrained

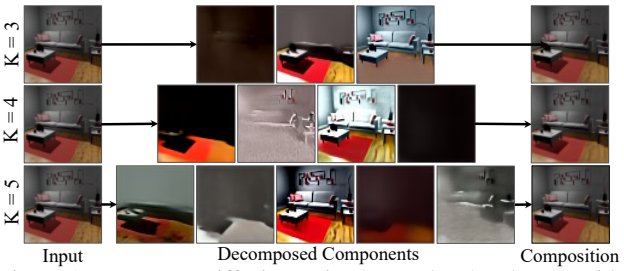

Figure 17: Decomp Diffusion trained on Falcor3D dataset with varying number of components $K = 3, 4$, and 5

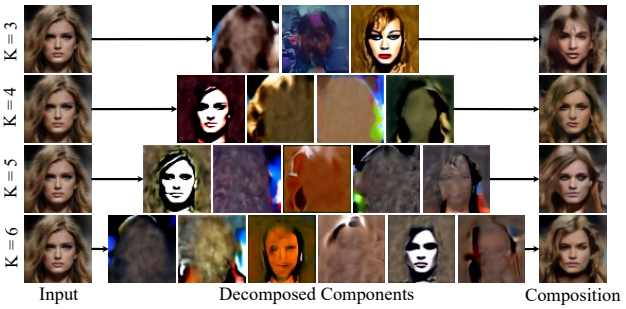

Figure 18: Decomp Diffusion trained on CelebA-HQ with varying number of components $K = 3, 4, 5$, and 6

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, *i.e.*, 1 residual block and convolution layer, as well as depth 2, depth 3 (the default value we used in the main paper), and depth 5, with results shown in Figure 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.

D Model Details

We used the standard U-Net architecture from (Ho et al., 2020) as our diffusion model. To condition on each inferred latent $\boldsymbol{z}_k$, we concatenate the time embedding with encoded latent $\boldsymbol{z}_k$, 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 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.

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

Table 4: Training dataset sizes.

E Experiment Details

In this section, we first provide dataset details in Section E.1. We then describe training details for our baseline methods in Section E.2. Finally, we present training and inference details of our method in Section E.3 and Section E.4.

E.1 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 and typically includes all available images unless specified otherwise.

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 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.

Anime. (Branwen et al., 2019) When creating the multi-modal faces dataset, we combined a 30,000 cropped Anime face images with 30,000 CelebA-HQ images.

Tetris. (Greff et al., 2019) We used a smaller subset of 10K images in training, due to the simplicity of the dataset.

KITTI. (Geiger et al., 2012) We used 8,008 images from a scenario in the the Stereo Evaluation 2012 benchmark in our training.

Virtual KITTI 2. (Cabon et al., 2020) We used 21,260

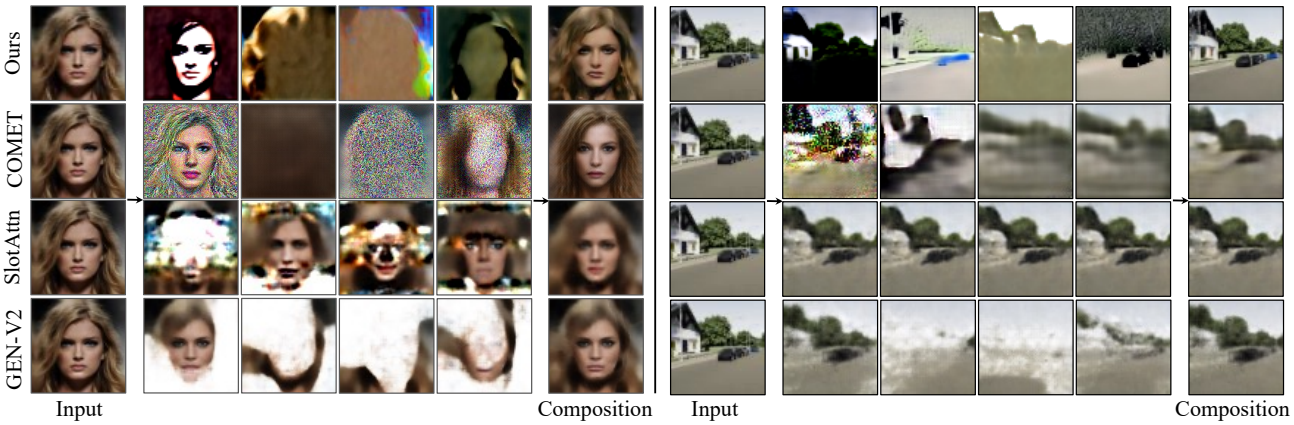

Figure 19: **Qualitative comparisons on CelebA-HQ and VKITTI datasets**. Decomposition results on CelebA-HQ (**Left**) and Virtual KITTI 2 (**Right**) on benchmark object representation methods. Compared to our method, COMET generates noisy components and less accurate reconstructions. SlotAttention may produce identical components, and it and GENESIS-V2 cannot disentangle global-level concepts.

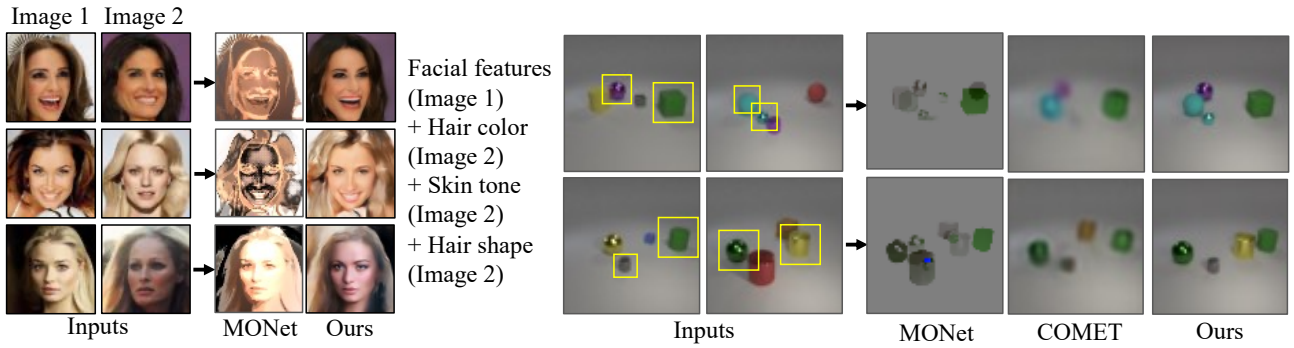

Figure 20: **Recombination comparisons on CelebA-HQ and CLEVR with MONet**. We further compare with MONet on recombination. Our method outperforms MONet by generating correct recombinations results.

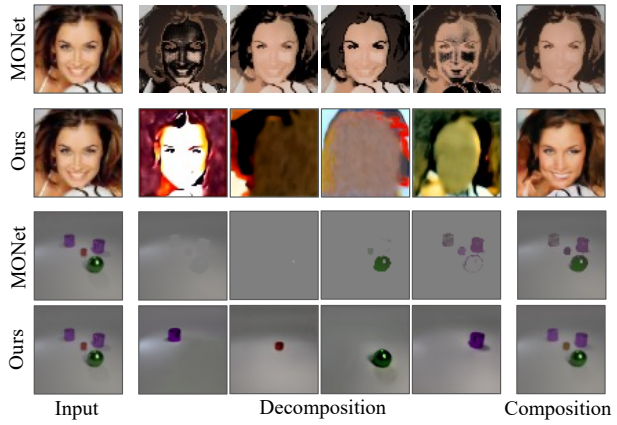

Figure 21: **Decomposition comparisons on CelebA-HQ and CLEVR datasets**. We provide qualitative comparisons on decomposition between MONet and our method. Our method can decompose images into factors that are more visually diverse and meaningful, while MONet may fail to disentangle factors.

images from a setting in different camera positions and weather conditions.

E.2 Baselines

Info-GAN (Chen et al., 2016). We train Info-GAN using the default training settings from the official codebase at https://github.com/openai/InfoGAN.

β-VAE (Higgins et al., 2017). 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 (Burgess et al., 2019). 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 (Du et al., 2021a). We use the official codebase to train COMET models on various datasets, with a default setting that utilizes 64 as the dimension for the latent variable $\boldsymbol{z}$. Each model is trained until convergence over a period of $100,000$ iterations. We use the codebase in https://github.com/yilundu/comet.

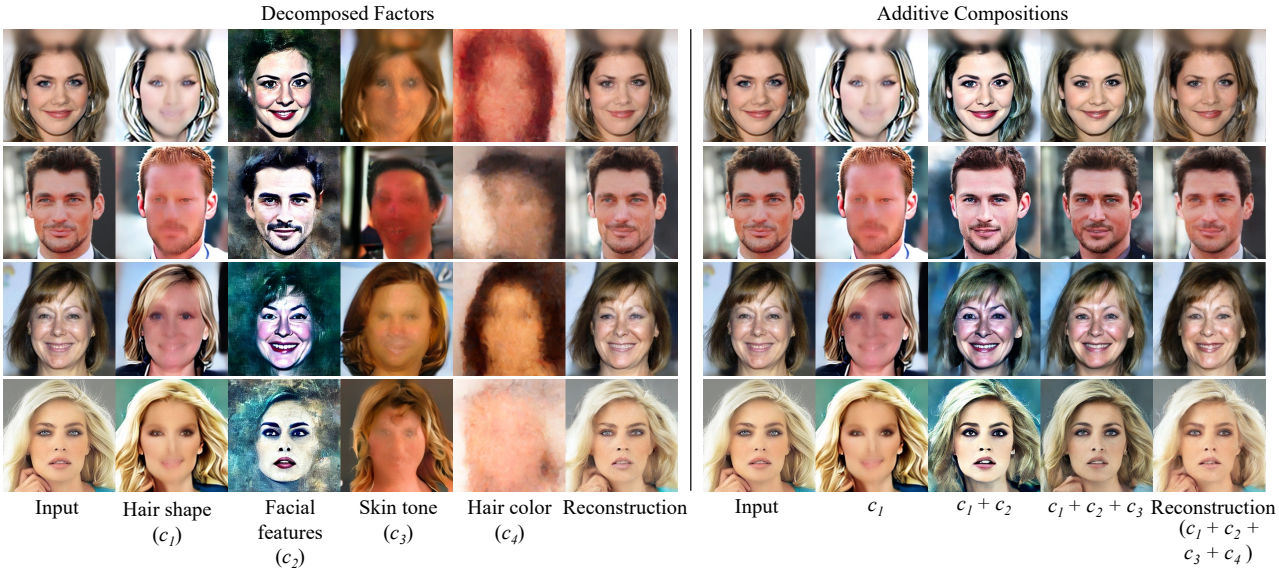

Figure 22: **Additive Factors Composition on CelebA-HQ.** On the left, we show decomposed components on CelebA-HQ images with inferred labels. On the right, we present compositions generated by adding one factor at a time to observe the information learned by each component.

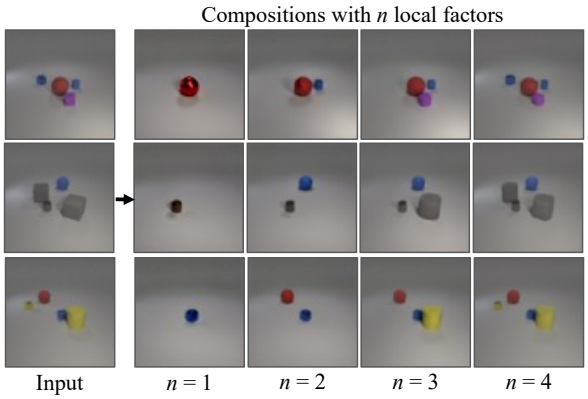

Figure 23: **Additive Factors Composition on CLEVR.** We demonstrate that each decomposed object factor can be additively composed to reconstruct the original input image.

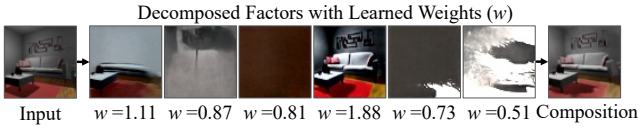

Figure 24: **Systematic Selection of Latent Set Size.** We can optionally learn weights for latent components during training. This approach is helpful for automatically choosing the number of components, as we can remove the most insignificant latent components based on their weights.

Slot Attention (Locatello et al., 2020b). We use an existing PyTorch implementation to train Slot Attention from https://github.com/evelinehong/slot-attention-pytorch .

GENESIS-V2 (Engelcke et al., 2021b). We train GENESIS-V2 using the default training settings from the official codebase at https://github.com/applied-ai-lab/genesis .

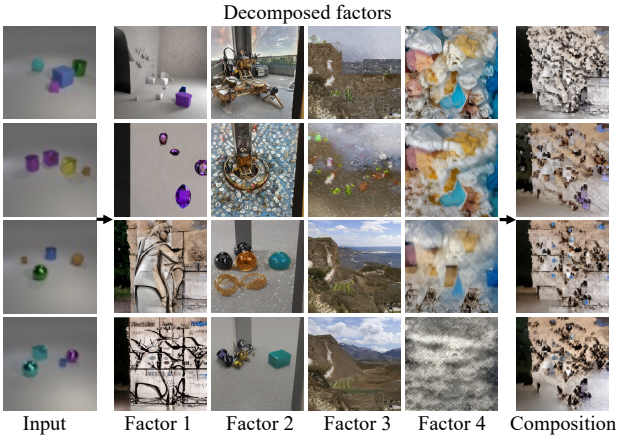

Figure 25: **One-Shot Decomposition using (Liu et al., 2023b).** The method fails to decompose objects in the input training image.

E.3 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 $\boldsymbol{x}_0$, 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 $\boldsymbol{z}_k$, we first utilize the latent encoder to encode input

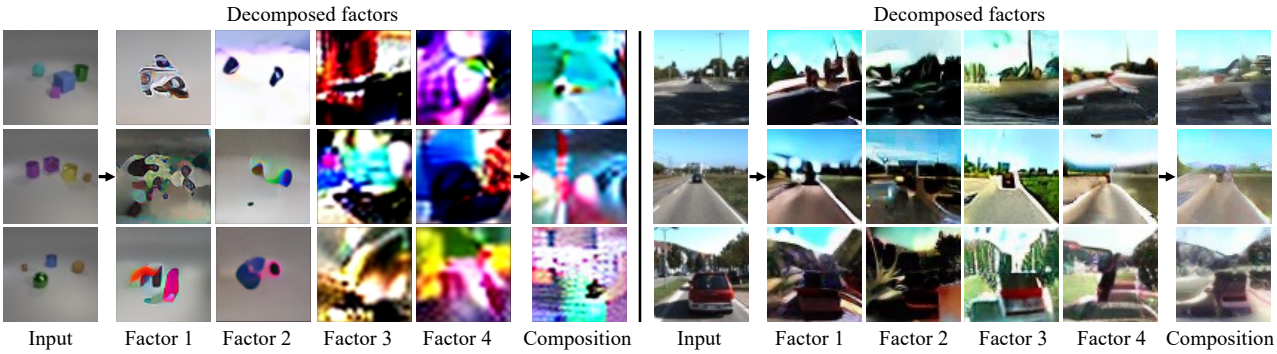

Figure 26: **Decomposition with Pretrained Stable Diffusion.** We find that applying our approach with pre-trained Stable Diffusion model doesn't not help find meaningful factors on both CLEVR and KITTI datasets.

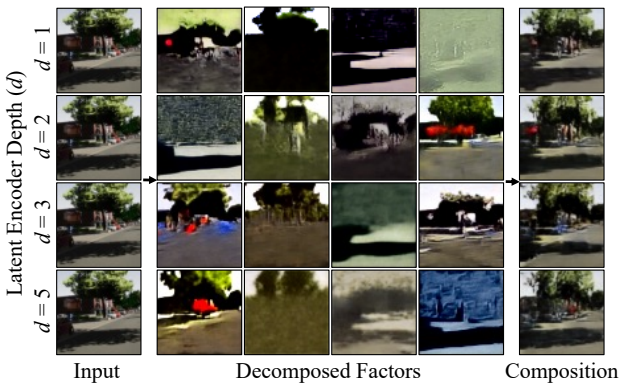

Figure 27: **Impact of latent encoder depth on VKITTI.** Encoders with different depths, denoted as d, can learn similar decomposed factors, including shadows, background, etc.

images into features that are further split into a set of latent representations $\{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K\}$. For each input image, we then train our model conditioned on each decomposed latent factor $\boldsymbol{z}_k$ 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.

E.4 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, *i.e.*, 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, *i.e.*, 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, *e.g.*, 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.

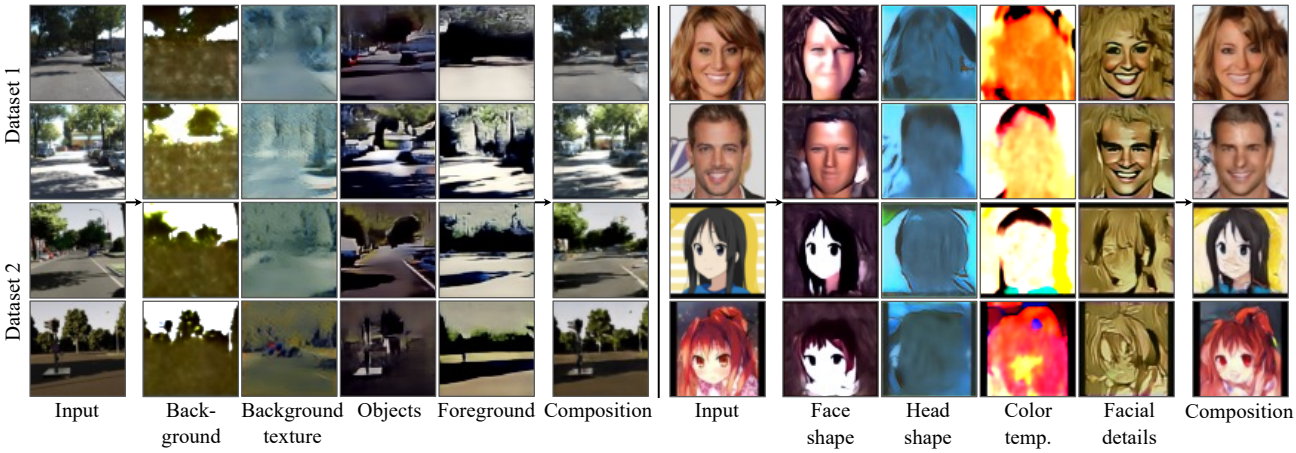

Figure 28: **Multi-modal Dataset Decomposition.** Multi-model decomposition and composition results on hybrid datasets such as KITTI and Virtual KITTI 2 scenes (**Left**), and CelebA-HQ and Anime faces (**Right**). The top 2 images are of the first dataset, and the bottom 2 images are of the second dataset. Inferred concepts are named for better understanding.

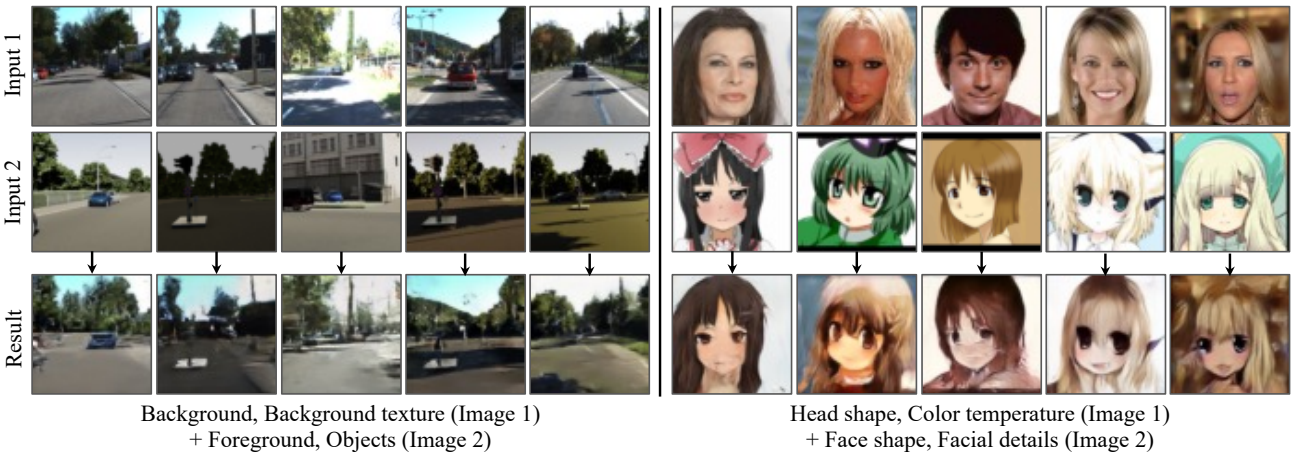

Figure 29: **Multi-modal Dataset Recombination.** Recombinations of inferred factors from hybrid datasets. We recombine different extracted factors to generate unique compositions of KITTI and Virtual KITTI 2 scenes (**Left**), and compositions of CelebA-HQ and Anime faces (**Right**).

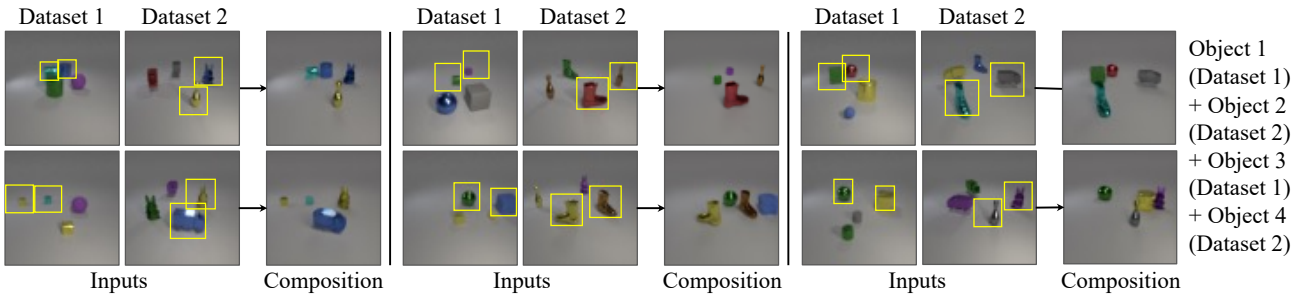

Figure 30: **Cross Dataset Recombination.** We further showcase our method's ability to recombine across datasets using 2 different models that train on CLEVR and CLEVR Toy, respectively. We compose inferred factors as shown in the bounding box from two different modalites to generate unseen compositions.