Learning to Compose Visual Relations

Anonymous Author(s)
Affiliation
Address
email

Abstract

The visual world around us can be concisely described as a structured set of objects and their associated relations. An image of a room may be conjured given only the description of the underlying objects and their associated relations. While there has been significant work on designing deep neural networks which may compose individual objects together, less work has been done on composing the individual relations between objects. A principal difficulty is that while the placement of objects is mutually independent, their relations are entangled and dependent on each other. To circumvent this issue, existing works primarily compose relations by utilizing a holistic encoder, in the form of text or graph. In this work, we instead propose to represent each relation as an unnormalized density (an energy-based model), enabling us to compose separate relations in a factorized manner. We show that such a factorized decomposition allows more faithfully to both generate and edit scenes that have multiple sets of relations. We further show that decomposition enables our model to effectively understand the underlying relational scene structure.

1 Introduction

The ability to reason about the component objects and their relations in a scene is key for a wide variety of robotics and AI tasks, such as multistep manipulation planning [12], concept learning [24], navigation [40], and dynamics prediction [2]. While a large body of work has explored inferring and understanding the underlying objects in a scene, robustly understanding the component relations in a scene remains a challenging task. In this work, we explore how to robustly understand relational scene description (Figure 1).

Naively, one approach towards understanding relational scene descriptions is to utilize existing multi-modal language and vision models. Such an approach has recently achieved great success in DALL-E [33] and CLIP [32], both of which show compelling results on encoding object properties with language. However, when these approaches are instead utilized to encode relations between objects, their performance rapidly deteriorates, as shown in [33] and which we further illustrate in Figure 6. We posit that an issue with these models is that the language encoder is not *compositional* with respect to the underlying relations in an image, thus they fail to consider every single relation that describes a scene.

To remedy this issue, we propose instead to *factorize* the scene description with respect to each individual relation. Separate models are utilized to encode each individual relation and are then subsequently *composed* together to represent a relational scene description. The most straightforward manner is to specify distinct regions of an image in which each relation can be located, with a composed relation description corresponding to combining all separate regions of an image together.

Such an approach has significant drawbacks. In practice, the location of one pair of objects in a relation description may be heavily influenced by the location of objects specified by another relation

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

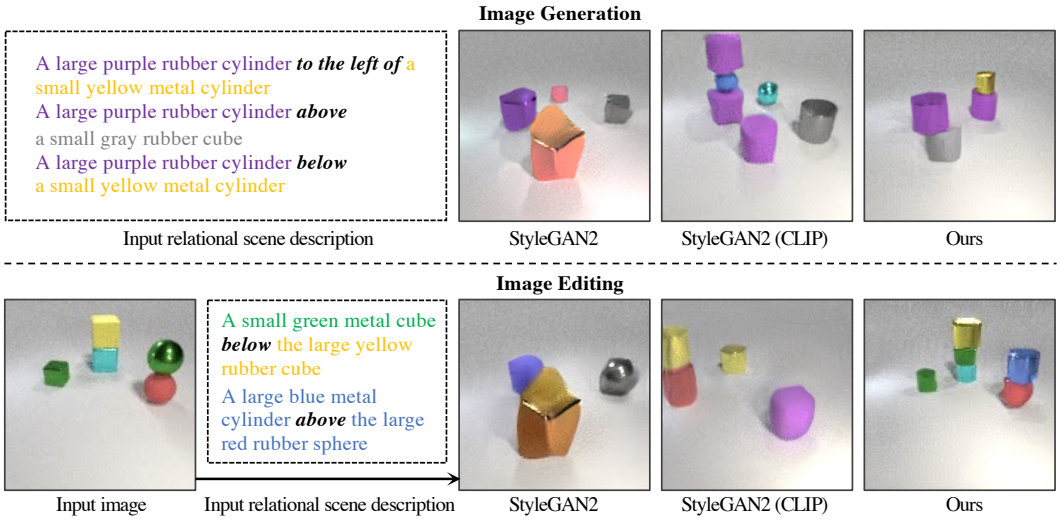

Figure 1: Our model can generate and edit images with multiple composed relations. **Top**: Image generation results base on a relational description. **Bottom**: Image editing results base on a relational description.

description. Determining a priori the exact location of a relation will thus severely hamper the number of possible scenes that can be realized by giving a set of relations. Is it possible to factorize relational description of scene yet simultaneously generate images that take each given relation description in account?

In this work, we propose to represent and factorize individual relations as unnormalized densities using Energy-Based Models. A relational scene description is represented as the product of the individual probability distributions across relations, with each individual relation specifying a separate probability distribution over images. Such a composition enables interactions between multiple relations to be modeled.

We show that this resultant framework enables us to reliably capture and generate images with multiple composed relational descriptions. It further enables us to edit images to have a desired set of relations. Finally, by measuring the relative densities assigned to different relational descriptions, we are able to infer the objects and their relations in a scene for downstream tasks, such as image-to-text retrieval and classification.

There are three main contributions of our work. First, we present a framework to factorize and compose separate object relations. We show that the proposed framework is able to generate and edit images with multiple composed relations and significantly outperforms baseline approaches. Secondly, we show that our approach is able to infer the underlying relational scene descriptions and is robust enough in understanding semantically equivalent relational scene descriptions. Finally, we show that our approach can generalize to a previously unseen relation description, even if the underlying objects and descriptions are from a separate dataset not seen during training. We believe that such generalization is crucial for a general artificial intelligence system to adapt to the infinite number of variations of the world around it.

2 Related Work

Language Guided Scene Generation. A large body of work has explored scene generation utilizing text descriptions [14, 18, 21, 28, 33, 35–37, 42, 44, 45]. In contrast to our work, prior work [18, 36, 44, 45] have focused on generating images given only a limited number of relation descriptions. Recently, [33] shows compelling results on utilizing text to generate images, but also explicitly state that relational generation was a weakness of the model. In this work, we seek to tackle how we may generate images given an underlying relational description.

Visual Relation Understanding. To understand visual relations in a scene, many works applied neural networks to graphs [1–3, 15, 16, 19, 21, 26, 27, 30, 34]. Raposo et al. [34] proposed Relation Network (RN) to explicitly compute relations from static input scenes while we implicitly encode relational information to generate and edit images based on given relations and objects. Johnson et al. [21] proposed to condition image generation on relations by utilizing scene graphs which were

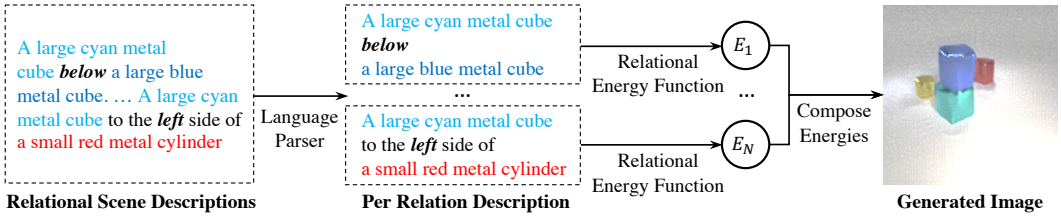

Figure 2: Overview of our pipeline for understanding a relational scene description. A relational scene description is parsed into a set of underlying relation descriptions. Individual relation descriptions are represented as EBMs which are subsequently composed together to generate an image.

further explored in [1, 15, 16, 19, 27, 29, 41]. In this setup, however, to condition on relations, the approaches require excessive supervisions, such as bounding boxes, segmentation masks, etc to infer and generate the underlying relations in an image. Such a setting greatly restricts the possible combinations of individual relations in a scene. In our work, we present a method that enables us to generate images given only a relational scene description.

Energy Based Models. Our work is related to existing work on energy-based models [5, 10, 11, 13, 23, 31, 39, 43]. Most similar to our work is that of [8], which proposes a framework utilizing EBMs to compose several object descriptions together. In contrast, in this work, we study the problem of how we may compose relational descriptions together, an important and challenging task for existing text description understanding systems.

3 Method

Given a training dataset $\mathcal{C} = \{\boldsymbol{x}_i, R_i\}_{i=1}^N$ with N distinct images $\boldsymbol{x}_i \in \mathbb{R}^D$ and associated relational descriptions $R_i := \{w_i^1, \dots w_i^n\}$, consisting of a set of n words w_i^j describing a set of K relations, we are interested in learning the underlying probability distribution $p_\theta(\boldsymbol{x}|R)$ — the probability distributions of an image given a corresponding relational description. To represent $p_\theta(\boldsymbol{x}|R)$, we split a relational description R into its K separate relations $\{r^k\}$ and model each component relation separately using a probability distribution $p_\theta(\boldsymbol{x}|r^k)$ each represented as an Energy-Based Model. Our overall scene probability distribution is then modeled by a composition of individual probability distributions for each relation description $p_\theta(\boldsymbol{x}|R) = \prod_k p_\theta(\boldsymbol{x}|r^k)$.

In this section, we give an overview of our approach towards factorizing and representing a relational scene description. We first provide a background overview of Energy-Based Models (EBMs) in Section 3.1. We then describe how we may parameterize individual relational probability distributions with EBMs in Section 3.2. We further describe how we compose relational probability distributions to model a relational scene description in Section 3.3. Finally, we illustrate downstream applications of our relational scene understanding model in Section 3.4.

3.1 Energy-Based Models

We model each relational probability distribution utilizing an Energy-Based Model (EBM) [6, 25]. EBMs are a class of unnormalized probability models, which parameterize a probability distribution $p_\theta(\boldsymbol{x}$ utilizing using a learned energy function $E_\theta(\mathbf{x})$, represented by a neural network,

$$p_\theta(\mathbf{x}) \propto e^{-E_\theta(\mathbf{x})}. \tag{1}$$

EBMs are typically trained utilizing contrastive divergence [17], where energies of training datapoints are decreased and energies of sample data points from $p_\theta(\boldsymbol{x})$ are increased. We utilize the training code and models from [9] to train our EBMs. To generate samples from an EBM $p_\theta(\boldsymbol{x})$, we utilize MCMC sampling on the underlying distribution, and Langevin dynamics, which refines a data sample iteratively from a random noise as follows:

$$\tilde{\mathbf{x}}^k = \tilde{\mathbf{x}}^{k-1} - \frac{\lambda}{2}\nabla_{\mathbf{x}} E_\theta(\tilde{\mathbf{x}}^{k-1}) + \omega^k,\ \omega^k \sim \mathcal{N}(0, \lambda) \tag{2}$$

where k is the kth iteration and λ is the step size, utilizing the gradient of the energy function with respect to the input sample $\nabla_{\mathbf{x}} E_\theta$. EBMs enable us to naturally compose separate probability distributions together [7]. In particular, given a set of probability distributions $\{p_\theta^i(\boldsymbol{x})\}$, the joint

probability distribution is represented as:

$$\prod_i p_\theta^i(\mathbf{x}) \propto e^{-\sum_i E_\theta^i(\mathbf{x})}, \tag{3}$$

which we use Langevin dynamics to sample from.

3.2 Learning Relational Energy Functions

Given a scene relation r_i, described using a set of words $\{w_i^1, \dots w_i^n\}$, we seek to learn a conditional EBM which models the underlying probability distribution $p(\boldsymbol{x}|r_i)$.

$$p_\theta(\boldsymbol{x}; r_i) \propto e^{-E_\theta(\boldsymbol{x}; \text{Enc}(r_i))}, \tag{4}$$

where $\text{Enc}(r_i)$ denotes a text encoder for describing the relation.

The most straightforward manner in which we encode a relation is to utilize an existing text encoder, such as CLIP, for the entire sentence describing a relation. However, we find that such an approach, as illustrated in Figure 6, does not capture relations well, and we demonstrate in the supplement that it performs poorly. We posit that an issue with directly encoding a relation using its sentence encoder is that an insufficient weight is put on the underlying relation token in r_i

To enforce that the underlying relation token in r_i is effectively encoded, we instead propose to decompose the relation r_i using a parser into a relation triplet (r_i', o_i^1, o_i^2), corresponding to the relation token, object 1 description and object 2 description in r_i. Each separate entry in the relation triplet is then separately embedded.

We found that such an approach puts a suitable weight on the underlying relation token of a sentence and enables us to effectively model relation distributions. We explored two separate approaches to embed the underlying object descriptions into our relational energy function.

CLIP Embedding. One approach we consider is to directly utilize CLIP to obtain the embedding of an object description. Such an approach may potentially enable us to generalize relations in a zero-shot manner to new objects by utilizing CLIP's underlying embedding of the object, but may also hurt learning if the underlying object embedding does not distinctly separate two object descriptions.

Random Initialization. Alternatively, we may directly encode an object description by utilizing randomly initialized embeddings. In this approach, we extract a scene embedding by concatenating the learned embeddings of color, shape, material, size of o_i^1, o_i^2 and their relation r_i'.

3.3 Representing Relational Scene Descriptions

Given an underlying scene description R, we represent the underlying probability distribution $p(\boldsymbol{x}|R)$ by factorizing it as a product of probabilities over the the underlying relations r_k inside R. Given the separate relational energy functions learned in Section 3.2, this probability is proportional to

$$p(\boldsymbol{x}|R) = e^{-E_\theta(\boldsymbol{x}|R)} = \prod_k p(\boldsymbol{x}|r_k) = e^{-\sum_k E_\theta(\boldsymbol{x}|r_k)}, \tag{5}$$

which is a new EBM with underlying energy function $E_\theta(\boldsymbol{x}|R) = \sum_k E_\theta(\boldsymbol{x}|r_k)$. We provide an overview of this setup in Figure 2, where $E_1, \cdots, E_N$ correspond to the N individual relational energy functions $E_\theta(\boldsymbol{x}|r_k)$.

3.4 Downstream Applications

By learning the probability distribution $p(\boldsymbol{x}|R)$ with corresponding EBM $E_\theta(\boldsymbol{x}|R)$, representing a probability distribution over images under a relation description R, our model can be applied to downstream applications, such as image generation, editing, and classification, which we detail below and validate in the experiment section.

Image Generation. We generate images from a relational scene description R by sampling from the probability distribution $p(\boldsymbol{x}|R)$ using Langevin sampling on the energy function $E_\theta(\boldsymbol{x}|R)$.

Image Editing. We further utilize $p(\boldsymbol{x}|R)$ to edit an image $\boldsymbol{x}'$. We define a relational probability distribution conditioned on the edited image $p(\boldsymbol{z}|R, \boldsymbol{x}') = p(\boldsymbol{x}|R)\mathcal{N}(\boldsymbol{x}', \sigma)$, which contains an additional term centered at $\boldsymbol{x}'$ to encourage generated images to be near $\boldsymbol{x}'$. We utilize Langevin sampling on this modified probability distribution, initialized from our original image $\boldsymbol{x}'$, to edit $\boldsymbol{x}'$ to satisfy a scene relational scene description R.

Table 1: Evaluation of the accuracy of object relations in the generated images or edited images on the Clevr and iGibson datasets. We compare our method with baselines on three test setsets, *i.e. 1R*, *2R*, and *3R* (see text).

Dataset	Model	Image Generation (%)			Image Editing (%)		
		1R Acc	2R Acc	3R Acc	1R Acc	2R Acc	3R Acc
Clevr	StyleGAN2	10.64	2.20	0.66	14.30	3.20	1.20
	StyleGAN2 (CLIP)	58.62	11.48	3.42	-	-	-
	Ours (CLIP)	92.46	57.06	26.68	98.40	**42.30**	**15.20**
	Ours (Learned Embedding)	**95.30**	**59.42**	**27.58**	**99.20**	41.50	13.50
iGibson	StyleGAN2	13.10	3.30	1.00	12.98	2.80	0.98
	StyleGAN2 (CLIP)	89.70	12.60	3.70	-	-	-
	Ours (CLIP)	**94.00**	**46.30**	**17.70**	**97.80**	**28.10**	**9.10**
	Ours (Learned Embedding)	81.40	27.50	9.70	86.00	22.70	8.40

Relational Scene Understanding. We may further utilize our energy function $E_\theta(\boldsymbol{x}|R)$ as a tool for relational scene understanding by noting that $p(\boldsymbol{x}|R) \propto e^{-E_\theta(\boldsymbol{x}|R)}$, and thus using the output values of the energy function as a ranking tool to separate relational scene description pairs.

4 Experiment

We conduct empirical studies to answer the following questions: (1) Can we learn relational models that can generate and edit complex multi-object scenes when given relational scene descriptions with multiple composed scene relations? (2) Can we use our model to generalize to scenes that are never seen in training? (3) Can we understand the nested relations in a scene and infer semantically equivalent descriptions?

To answer these questions, we evaluate the proposed method and baselines on image generation, image editing, and image classification on two datasets, *i.e.* Clevr [20] and iGibson [38].

4.1 Datasets

Clevr. We use $50,000$ pairs of images and relational scene descriptions for training. Each image contains $1 \sim 5$ objects and each object consists of five different attributes, including color, shape, material, size, and its spatial relation to another object in the same image. There are 9 types of colors, 4 types of shapes, 3 types of materials, 3 types of sizes, and 6 types of relations.

iGibson. On the iGibson dataset, we also use 50,000 pairs of images and relational scene descriptions for training. Each image contains $1 \sim 3$ objects and each object consists of the same five different types of attributes as the Clevr dataset. There are 6 types of colors, 5 types of shapes, 4 types of materials, 2 types of sizes, and 4 types of relations. The objects are randomly placed in the scenes.

In the training set, each image's relational scene description only contains one scene relation and objects are randomly placed in the scene. We generated three test subsets that contain relational scene descriptions with a different number of scene relations to test the generation ability of the proposed methods and baselines. The *1R* test subset is similar to the training set where each relational scene description contains one scene relation. The *2R* and *3R* test subsets have two and three scene relations in each relational scene description, respectively. Each test set has $5,000$ images with corresponding relational scene descriptions.

4.2 Baselines

We compare our method with two baseline approaches. The first baseline is StyleGAN2 [22], one of the state-of-the-art methods for unconditional image generation. To enable StyleGAN2 to generate images and edit images based on relational scene descriptions, we train a ResNet-18 classifier on top of it to predict the object attributes and their relations. Recently, CLIP [32] has achieved a substantial improvement on the text-image retrieval task by learning good text-image feature embedding on large-scale datasets. Thus we design another baseline, StyleGAN2+CLIP, that combines the best approach from both the image generation and text-image side for comparison. To do this, we encode

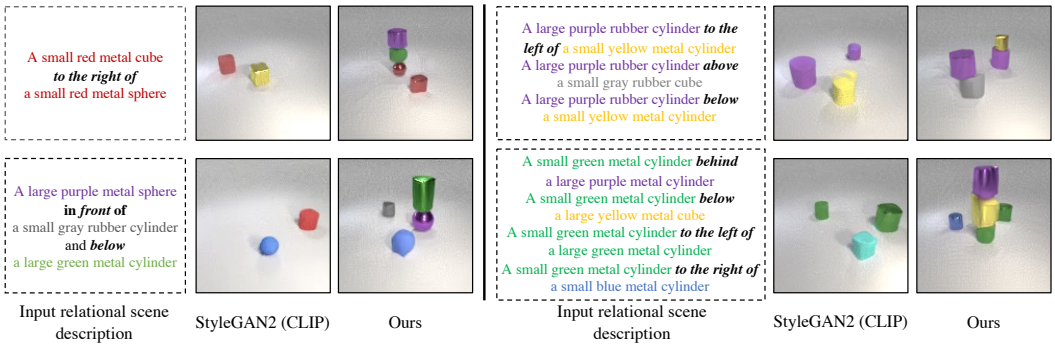

Figure 3: Examples of image generation on Clevr scenes. We show the image generation results based on $1 \sim 4$ relational descriptions. Note that the composed scene relations (2, 3, and 4 relational descriptions) are outside the training distribution and our approach can still generate correct images.

Figure 4: Examples of image generation on Clevr scenes. We show the image generation results based on $1 \sim 2$ relational descriptions. Note that the two composed scene relations are outside the training distribution and our approach can still generate correct images.

relational scene descriptions into text embedding using CLIP and condition on the embedding to generate images. Please see the **Supplement** for details of our approach and baselines.

4.3 Image Generation

Given a relational scene description, such as "a small blue cube on top of a red sphere", we aim at generating images that contain corresponding objects and their relations as described in the given descriptions. The proposed model and baselines generate images from Gaussian noise.

Quantitative comparisons. To evaluate the quality of generated images, we train a binary classifier to predict whether the generated image contains objects and their relations described in the given relational scene description. Given a pair of an image and a relational scene description, we first feed the image to several convolutional layers to generate an image feature and then send the relational scene description to an embedding layer followed by several fully connected layers to generate a relational scene feature. The image feature and relational scene feature are combined and then passed through several fully connected and finally a sigmoid function to predict whether the given image matches the relational scene description. The binary classifier is trained on real images from the training dataset. We train a classifier on each dataset and observe classification accuracy on real images to be close to 100%, indicating that the classifier is effective. During testing, we generate an image based on a relational scene description and send the generated image and the relational scene description to the classifier for prediction. For a fair comparison, we use the same classifier to evaluate images generated by all the approaches on each dataset.

The "Image Generation" column in Table 1 shows the classification results of different approaches on the Clevr and iGibson datasets. On each dataset, we test each method on three test subsets, *i.e.* *1R*, *2R*, *3R*, and report their binary classification accuracies. Both variants of our proposed approach outperform StyleGAN2 and StyleGAN2 (CLIP), indicating that our method can generate images that contain the objects and their relations described in the relational scene descriptions. We find that our approach using learned embedding achieves a better performance on the Clever dataset, while the other one using the CLIP embedding is better on the iGibson dataset.

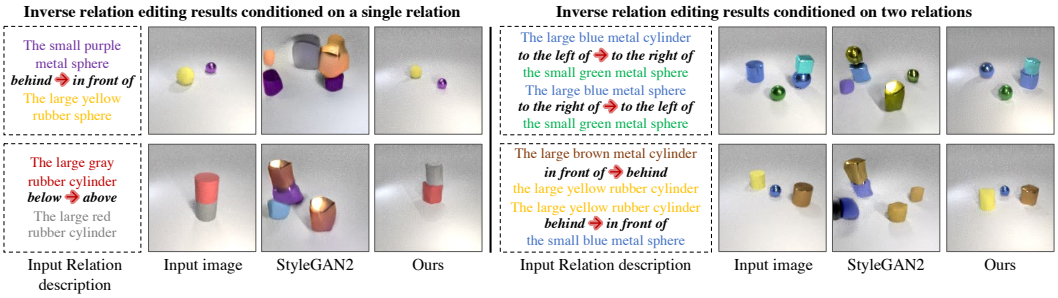

Figure 5: Examples of relation editing on Clevr scenes. **Left**: relation editing based on a single scene relation. **Right**: relation editing based on two composed scene relations. Note that the composed scene relations in the right part are outside the training distribution and our approach can still edit the images correctly.

StyleGAN2 and StyleGAN2 (CLIP) can perform well on the *1R* test subset. This is an easier test subset because the models are trained on images with a single scene relation and the models generate images based on the relational scene description during testing as well. The *2R* and *3R* are more challenging test subsets because the models need to generate images conditioned on relational scene descriptions of multiple scene relations. Our models outperform the baselines by a large margin, indicating the proposed approach has a better generalization ability and can compose multiple relations that are never seen during training.

Qualitative comparisons. The image generation results on Clevr and iGibson scenes are shown in Figure 3 and Figure 4, resepctively. We show examples of generated images conditioned on relational scene descriptions of different number of scene relations. Our method can generate correct images based on the relational scene descriptions. Note that both the proposed method and the baselines are trained on images that only contain a relational scene description of a single scene relation describing the visual relationship between two objects in each image. We find that our approach can still generalize well when composing more visual relations. Taking the upper right figure in Figure 3 as an example, a relational scene description of multiple scene relations, "A red leather couch to the right of a blue fabric couch. A red leather couch behind a maple wood cabinet", is never seen during training. StyleGAN2 (CLIP) generates wrong objects and scene relations that are different from the scene descriptions. In contrast, our method has the generalization ability to novel relational scenes.

4.4 Image Editing

Given an input image, we aim to edit this image based on relational scene descriptions, such as "put a red cube in front of the blue cylinder".

Quantitative comparisons. Similar to the image generation task, we use a classifier to predict whether the image after editing contains the objects and their relations in the relational scene description. We use the same classifier for both image generation and image editing on each dataset.

The "Image Editing" column in Table 1 shows the classification results of different approaches on the Clevr and iGibson datasets. Both variants of our proposed approach outperform the StyleGAN2 baseline substantially. The good performance of our approach on the *2R* and *3R* test subsets shows that the proposed method has a good generalization ability to relational scene descriptions that are outside the training distribution. The images after editing based on relational scene descriptions can incorporate the described objects and their relations correctly.

Qualitative comparisons. We show image editing examples in Figure 5. The left part is image editing results conditioned on relational scene descriptions of a single scene relation while the right part is conditioned on relational scene description of two scene relations. We show examples that edit images by inverting individual spatial relations between given two objects. Taking the first image in Figure 5 as an example, "The small purple metal sphere" is behind "The large yellow rubber sphere", after editing, our model can successfully put "The small purple metal sphere" in front of "The large yellow rubber sphere". Even for relational scene descriptions of two scene relations that are never seen during training, our model can edit images so that the selected objects are placed correctly.

4.5 Relational Understanding

We hypothesize the good generation performance of our proposed approach is because our approach understands the underlying nested relations and can distinguish different relational scene descriptions.

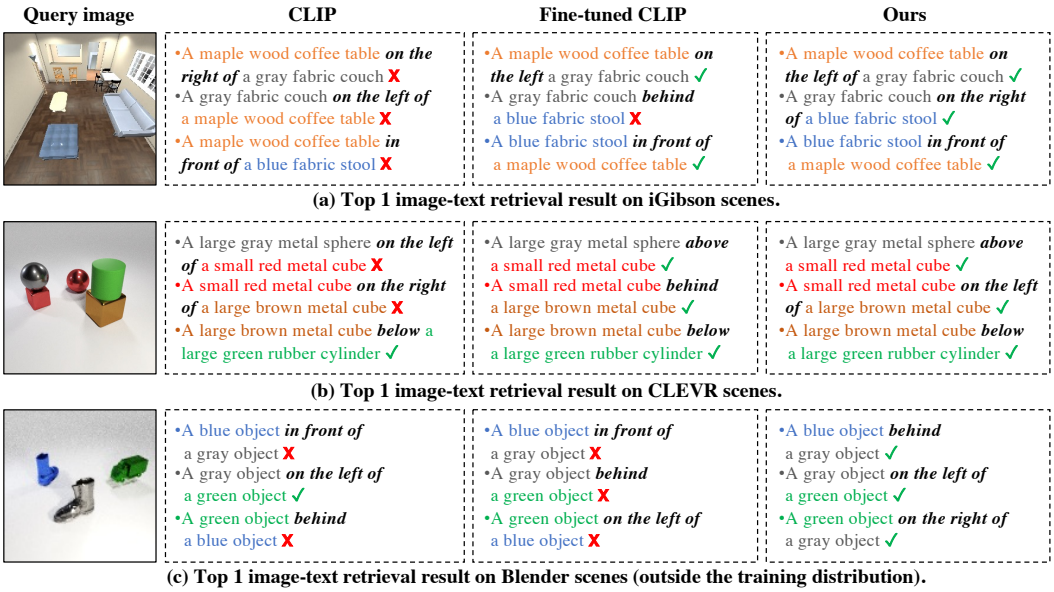

Figure 6: Image-to-text retrieval results. We compare the proposed approach with the pretrained CLIP and fine-tuned CLIP and show their top-1 retrieved relation description based on the given image query.

In this section, we evaluate the relational understanding ability of our proposed method and baselines by comparing their image-to-text retrieval and semantic equivalence results.

Image-to-text retrieval. In Figure 6, we evaluate whether our proposed model can understand different relational scene descriptions by image-to-text retrieval. We create a test set that contains 240 pairs of images and relational scene descriptions. Given a query image, we compute the similarity of this image and each relational scene description in the gallery set. The top 1 retrieved relational scene description is shown in Figure 6. We compare our method with two baselines. We use the released CLIP model and test it on our dataset. Fine-tuned CLIP is the CLIP model fine-tuned on our dataset. Even though CLIP has shown good performance on the general image-text retrieval task, we find that it cannot understand spatial relations well, while EBMs can retrieve all ground truth descriptions.

We also find that our approach generalizes across datasets. In the bottom row of Figure 6, we conduct an additional image-to-text retrieval experiment on the Blender [4] scenes that are never seen during training. Our approach can still find the correct relational scene description for the query image.

Can we understand semantically equivalent relational scene descriptions? Given two relational scene descriptions describing the same image but in different ways, can our approach understand that the descriptions are semantically similar or equivalent? To evaluate this, we create a test subset that contains $5,000$ images and each image has 3 different relational scene descriptions. There are two relational scene descriptions that match the image but describe the image in different ways, such as "a cabinet in front of a couch" and "a couch behind a cabinet". There is one further description that does not match the image. The relative score difference between the two ground truth relational scene descriptions should be smaller than the difference between one ground truth relational scene description and one wrong relational scene description.

Table 2: Quantitative evaluation of **semantic equivalence** of on Clevr dataset. It measures the relative difference between two pairs of label descriptions, including a pair of semantically equivalent labels and an opposite pair (a ground truth label and an incorrect label).

Model	**Semantic Equivalence (%)**		
	1R Acc	2R Acc	3R Acc
Classifier	60.34	15.36	6.18
CLIP	55.98	35.74	25.74
CLIP (Fine-tuned)	48.58	36.22	28.74
Ours (CLIP)	80.46	33.96	21.04
Ours (Learned Emb)	**85.34**	**45.66**	**32.62**

We compare our approach with three baselines. For each model, given an image, if the difference between two semantically equivalent relational scene description is smaller than the difference between the semantically different ones, we will classify it as correct. We compute the percentage of

Figure 7: Example of **semantic equivalence** on Clevr and iGibson scenes. Given an input image, our approach is able to recognize whether the relational scene descriptions are semantically equivalent or not.

correct predictions and show the results in Table 2. Our proposed method outperforms the baselines substantially, indicating that our EBMs can distinguish semantically equivalent relational scene descriptions and semantically nonequivalent relational scene descriptions. We further show two examples of our approach in Figure 7 from the iGibson and Clevr datasets. The energy difference between the semantically equivalent relational scene description is smaller than the mismatching pair.

4.6 Zero-shot Generalization Across Datasets

We find that our method can generalize across datasets as shown in the third example in Figure 6. We render a new relational dataset using objects such as boots, toys, and trucks. To quantitatively evaluate the generalization ability across datasets, we test the image-to-text retrieval accuracy on the Blender dataset. Note that our model and baselines are trained on Clevr and have never seen the Blender scenes during training.

We generate a Blender test set that contains 300 pairs of images and relational scene descriptions. For each image, we do text retrieval on the 300 relational scene descriptions. The top 1 accuracy is shown in Figure 8. We compare our approaches with two baselines, *i.e.* CLIP and CLIP fine-tuned on the Clevr dataset. We find the CLIP model and our approach using the CLIP embedding perform badly on the Blender dataset. This is because CLIP is not good at modeling relational scene description as we have shown in Section 4.5. Our approach using the learned embedding outperforms other methods, indicating that our EBMs with a good embedding feature can generalize well even on unseen datasets, such as Blender.

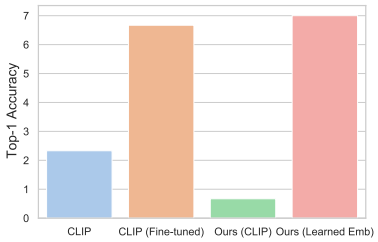

Figure 8: **Zero-shot generalization** on Blender scenes. Our approach with learned embeddings has higher top-1 accuracy than baselines on the image-to-text retrieval task.

5 Conclusion

In this paper, we demonstrate the potential usage of our models across tasks of compositional multi-object relational image generation, editing, and even generalization on unseen datasets. We believe our results provide evidence that EBMs are an interesting class of models to explore relational understanding.

A limitation of our current approach is that our evaluated datasets and settings are simple compared to the complex relational descriptions found in the real world. While we believe that our setting enables controlled and detailed evaluations of relational understanding, a good direction for future work would be to study how these models scale to complex datasets found in the real world. Of particular interest could be to measure the zero-shot relational generalization capabilities of the resultant model.

Our system, as with all systems based on neural networks, is susceptible to dataset biases. Our learned relations are obtained from supervised data/image relational scene description pairs, and thus will exhibit a bias if there is an underlying dataset bias. Care must be put in ensuring a balanced and fair dataset if our model is deployed in practice, as otherwise, it could inadvertently worsen existing societal prejudices and biases.

References

[1] Oron Ashual and Lior Wolf. Specifying object attributes and relations in interactive scene generation, 2019.

[2] Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics, 2016.

[3] Nicola De Cao and Thomas Kipf. Molgan: An implicit generative model for small molecular graphs, 2018.

[4] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. URL `http://www.blender.org`.

[5] Yilun Du and Igor Mordatch. Implicit generation and generalization in energy-based models. *arXiv preprint arXiv:1903.08689*, 2019.

[6] Yilun Du and Igor Mordatch. Implicit generation and generalization in energy-based models, 2020.

[7] Yilun Du, Shuang Li, and Igor Mordatch. Compositional visual generation and inference with energy based models. In *NeurIPS 2020*, 2020.

[8] Yilun Du, Shuang Li, and Igor Mordatch. Compositional visual generation with energy based models. In *Advances in Neural Information Processing Systems*, 2020.

[9] Yilun Du, Shuang Li, Joshua Tenenbaum, and Igor Mordatch. Improved contrastive divergence training of energy based models. *arXiv preprint arXiv:2012.01316*, 2020.

[10] Yilun Du, Shuang Li, Joshua B Tenenbaum, and igor Mordatch. Improved contrastive divergence training of energy based models. In *Proceedings of the 38th international conference on Machine learning*. ACM, 2021.

[11] Ruiqi Gao, Erik Nijkamp, Diederik P Kingma, Zhen Xu, Andrew M Dai, and Ying Nian Wu. Flow contrastive estimation of energy-based models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7518–7528, 2020.

[12] Caelan Reed Garrett, Rohan Chitnis, Rachel Holladay, Beomjoon Kim, Tom Silver, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Integrated task and motion planning, 2020.

[13] Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019.

[14] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation, 2015.

[15] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene graph generation with external knowledge and image reconstruction, 2019.

[16] Roei Herzig, Amir Bar, Huijuan Xu, Gal Chechik, Trevor Darrell, and Amir Globerson. Learning canonical representations for scene graph to image generation, 2020.

[17] Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Comput.*, 14(8):1771–1800, August 2002. ISSN 0899-7667. doi: 10.1162/089976602760128018. URL `https://doi.org/10.1162/089976602760128018`.

[18] Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. Inferring semantic layout for hierarchical text-to-image synthesis, 2018.

[19] Tianyu Hua, Hongdong Zheng, Yalong Bai, Wei Zhang, Xiao-Ping Zhang, and Tao Mei. Exploiting relationship for complex-scene image generation, 2021.

[20] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning, 2016.

[21] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs, 2018.

[22] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data, 2020.

[23] Taesup Kim and Yoshua Bengio. Deep directed generative models with energy-based probability estimation. *arXiv preprint arXiv:1606.03439*, 2016.

[24] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015. ISSN 0036-8075. doi: 10.1126/science.aab3050. URL `https://science.sciencemag.org/content/350/6266/1332`.

[25] Yann LeCun, Sumit Chopra, Raia Hadsell, Marc'Aurelio Ranzato, and Fu-Jie Huang. A tutorial on energy-based learning. In G. Bakir, T. Hofman, B. Schölkopf, A. Smola, and B. Taskar, editors, *Predicting Structured Data*. MIT Press, 2006.

[26] Jianan Li, Jimei Yang, Aaron Hertzmann, Jianming Zhang, and Tingfa Xu. Layoutgan: Generating graphic layouts with wireframe discriminators, 2019.

[27] Yikang Li, Tao Ma, Yeqi Bai, Nan Duan, Sining Wei, and Xiaogang Wang. Pastegan: A semi-parametric method to generate image from scene graph, 2019.

[28] Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Generating images from captions with attention, 2016.

[29] Gaurav Mittal, Shubham Agrawal, Anuva Agarwal, Sushant Mehta, and Tanya Marwah. Interactive image generation using scene graphs, 2019.

[30] Nelson Nauata, Kai-Hung Chang, Chin-Yi Cheng, Greg Mori, and Yasutaka Furukawa. Housegan: Relational generative adversarial networks for graph-constrained house layout generation, 2020.

[31] Erik Nijkamp, Mitch Hill, Tian Han, Song-Chun Zhu, and Ying Nian Wu. On the anatomy of mcmc-based maximum likelihood learning of energy-based models. *arXiv preprint arXiv:1903.12370*, 2019.

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[33] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021.

[34] David Raposo, Adam Santoro, David Barrett, Razvan Pascanu, Timothy Lillicrap, and Peter Battaglia. Discovering objects and their relations from entangled scene representations, 2017.

[35] Scott Reed, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. Learning what and where to draw, 2016.

[36] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis, 2016.

[37] Scott Reed, Aäron van den Oord, Nal Kalchbrenner, Sergio Gómez Colmenarejo, Ziyu Wang, Dan Belov, and Nando de Freitas. Parallel multiscale autoregressive density estimation, 2017.

[38] Bokui Shen, Fei Xia, Chengshu Li, Roberto Martın-Martın, Linxi Fan, Guanzhi Wang, Shyamal Buch, Claudia D'Arpino, Sanjana Srivastava, Lyne P Tchapmi, Kent Vainio, Li Fei-Fei, and Silvio Savarese. igibson, a simulation environment for interactive tasks in large realistic scenes. *arXiv preprint*, 2020.

[39] Yunfu Song and Zhijian Ou. Learning neural random fields with inclusive auxiliary generators. *arXiv preprint arXiv:1806.00271*, 2018.

[40] Michael Stilman and James Kuffner. Navigation among movable obstacles: Real-time reasoning in complex environments. *International Journal of Humanoid Robotics*, 2(4):479–504, December 2005.

[41] Subarna Tripathi, Anahita Bhiwandiwalla, Alexei Bastidas, and Hanlin Tang. Using scene graph context to improve image generation, 2019.

[42] Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with pixelcnn decoders, 2016.

[43] Jianwen Xie, Yang Lu, Song-Chun Zhu, and Yingnian Wu. A theory of generative convnet. In *International Conference on Machine Learning*, pages 2635–2644, 2016.

[44] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks, 2017.

[45] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 5.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See supplemental material.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See supplemental material.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]