

Image Generation from Hyper Scene Graphs with Trinomial Hyperedges Using Object Attention

Miyake, Ryosuke

Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University

Matsukawa, Tetsu

Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University

Suzuki, Einoshin

Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University

<https://hdl.handle.net/2324/7357487>

出版情報 : 2, pp.266-279, 2024. Science and Technology Publications

バージョン :

権利関係 : © 2024 by SCITEPRESS - Science and Technology Publications, Lda.



Image Generation from Hyper Scene Graphs with Trinomial Hyperedges Using Object Attention

Ryosuke Miyake, Tetsu Matsukawa^a and Einoshin Suzuki^b

Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University, Fukuoka, Japan

Keywords: Image Generation, Hyper Scene Graph, Object Attention.

Abstract: Conditional image generation, which aims to generate consistent images with a user's input, is one of the critical problems in computer vision. Text-to-image models have succeeded in generating realistic images for simple situations in which a few objects are present. Yet, they often fail to generate consistent images for texts representing complex situations. Scene-graph-to-image models have the advantage of generating images for complex situations based on the structure of a scene graph. We extended a scene-graph-to-image model to an image generation model from a hyper scene graph with trinomial hyperedges. Our model, termed hsg2im, improved the consistency of the generated images. However, hsg2im has difficulty in generating natural and consistent images for hyper scene graphs with many objects. The reason is that the graph convolutional network in hsg2im struggles to capture relations of distant objects. In this paper, we propose a novel image generation model which addresses this shortcoming by introducing object attention layers. We also use a layout-to-image model auxiliary to generate higher-resolution images. Experimental validations on COCO-Stuff and Visual Genome datasets show that the proposed model generates more natural and consistent images to user's inputs than the cutting-edge hyper scene-graph-to-image model.

1 INTRODUCTION

Conditional image generation, which aims to generate consistent images with a user's input, is one of the critical problems in computer vision. Text-to-image models (Reed et al., 2016; Zhang et al., 2017; Zhang et al., 2018; Odena et al., 2017) have been extensively studied in this problem because of their simple input and applicability in various fields. Although these models have succeeded in generating realistic images for simple situations with few objects, they often fail to generate consistent images for texts representing complex situations with multiple objects and their relationships.

This drawback stems from the difficulty of mapping a long sentence into a single feature vector. Scene-graph-to-image models can address this drawback by using the structured representation of scene graphs (Johnson et al., 2018). A scene graph consists of nodes representing objects and binomial edges describing relationships between two objects (Figure 2 (a)), enabling it to explicitly represent a complex

situation compared with text. Scene-graph-to-image models simplify the encoding of complex situations by converting each object into a feature vector. From these facts, these models expect to generate proper images for complex situations.

On the other hand, scene-graph-to-image models also have a shortcoming, i.e., inaccurate object positions. We addressed this shortcoming by proposing an image generation model from hyper scene graphs hsg2im (Miyake et al., 2023), as an extension of sg2im (Johnson et al., 2018). Hyper scene graphs include trinomial hyperedges representing positional relations among three objects. We increased the types of hyperedges (Miyake et al.,) of our previous work (Miyake et al., 2023). A trinomial hyperedge represents a positional relation among three objects. A trinomial hyperedge enables the model to process the three objects with one application of a Multi-Layer Perceptron (MLP), which makes capturing the relation among three objects easier.

However, hsg2im also has a shortcoming of generating an unnatural layout for a hyper scene graph with many objects. Figure 1 shows examples of this shortcoming. In the layout generated by our hsg2im (Miyake et al.,), the objects are neither arranged natu-

^a  <https://orcid.org/0000-0002-8841-6304>

^b  <https://orcid.org/0000-0001-7743-6177>

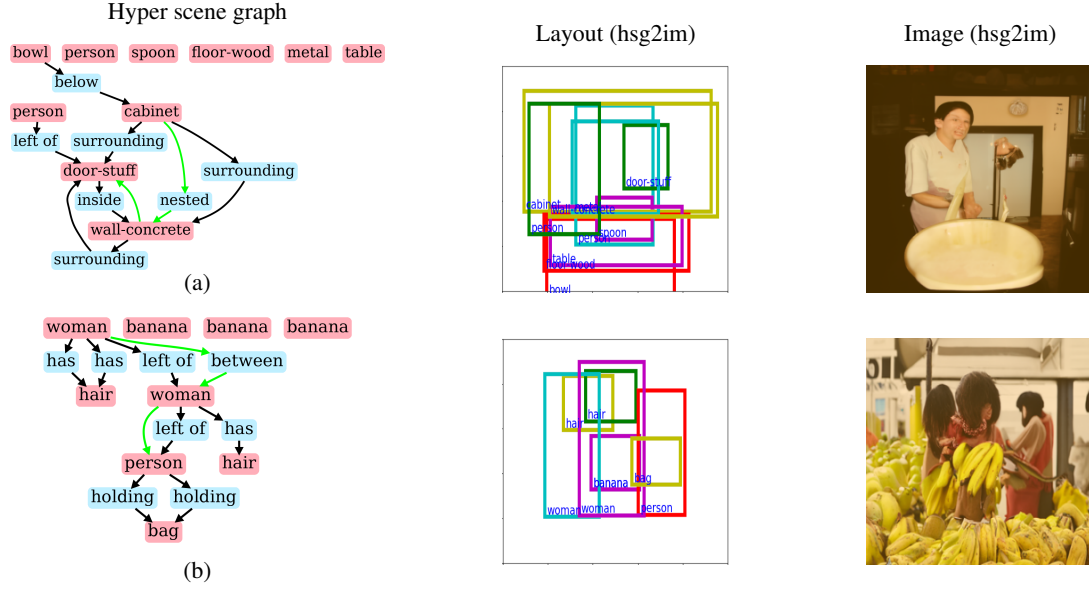


Figure 1: Examples showing the shortcoming of hsg2im (Miyake et al., 2023; Miyake et al.,).

rally nor consistently. In the example of (a), many objects, i.e., floor-wood (red box), table (purple), spoon (purple), and bowl (red) overlap each other in the middle center, and they cannot be seen in the generated image. In the example of (b), the hyper scene graph has a hyperedge woman (light blue) - between - woman (purple) $\rightarrow$ person (red); however, woman (light blue) and woman (purple) overlap each other in the layout generated by hsg2im, and we cannot see woman (light blue) clearly.

The reason lies in the graph convolutional network, which converts each object vector to reflect the other objects and relations in hsg2im. The graph convolutional network converts the feature vector based on the edges in the hyper scene graph. This process makes it difficult to generate object vectors that accurately reflect distant objects and relations. Consequently, hsg2im often fails to arrange the objects naturally for a hyper scene graph with many objects.

To address this shortcoming, we propose Object Attention hsg2im (OA-hsg2im). OA-hsg2im has self-attention layers for object vectors, which calculate the attention score for all combinations of two objects in a hyper scene graph. By converting object vectors with the attention scores, OA-hsg2im enables the objects to attend other objects relevant to themselves regardless of the distance in a hyper scene graph. This conversion also allows the object vectors to reflect a wider range of objects in the hyper scene graph. Therefore, OA-hsg2im is expected to generate natural layouts and images which are consistent with the input hyper scene graph even if it consists of many objects. In addition, we use a pre-trained layout-to-image model

LayoutDiffusion (Zheng et al., 2023) as an auxiliary model to generate higher-resolution images.

In this paper, we also tackle an investigation of prejudice in the image generation model. In recent years, the movements for eliminating such prejudice about genders or professions are becoming more active (Zhang et al., 2022). Vision and language datasets often reflect people’s prejudices (Tang et al., 2021), and image generation models could learn them. We investigate the prejudice about genders in the pre-trained LayoutDiffusion (Zheng et al., 2023).

2 RELATED WORK

We categorize conditional image generation models from three kinds of inputs; texts (Reed et al., 2016; Zhang et al., 2017; Zhang et al., 2018; Odena et al., 2017), layouts (Sun and Wu, 2019; He et al., 2021; Zheng et al., 2023; Hinz et al., 2022), and scene graphs (Johnson et al., 2018; Miyake et al., 2023; Miyake et al., ; Herzig et al., 2020). Text-to-image models have succeeded in generating realistic images for simple situations in which a few objects are present. Meanwhile, they often fail to generate consistent images for texts representing complex situations. Layout-to-image models and scene-graph-to-image models overcome this shortcoming of text-to-image models.

For layout-to-image models, He et al. proposed a model Layout2img, which generates consistent feature vectors for each object and generates natural images (He et al., 2021). Zhang et. al proposed a layout-

to-image diffusion model LayoutDiffusion, which has a diffusion architecture (Ho et al., 2020), enabling to generate higher-quality images than GAN-based methods (Zheng et al., 2023). Though layout-to-image models can control the position of the generated object with the input, they have difficulty in generating an image from a text due to the dissimilarity of the structures between a text and a layout.

For scene-graph-to-image models, Johnson et al. proposed a model sg2im (Johnson et al., 2018). Scene-graph-to-image models can generate proper images for complex situations due to the powerful structured representation of scene graphs (Johnson et al., 2018). Also, scene-graph-to-image models can be easily applied to text-to-image models due to the similarity of the structures between a text and a scene graph. Actually, Schuster et al. worked on the transformation from a text to a scene graph (Schuster et al., 2015).

Some researchers have attempted to generate more consistent images from scene graphs. Herzig et al. generated images from canonicalized scene graph (Herzig et al., 2020). Vo et al. introduced an auxiliary classifier loss in terms of the binomial relations (Vo and Sugimoto, 2020). We proposed hsg2im, which is an image generation model from a hyper scene graph with trinomial hyperedges (Miyake et al., 2023), by extending sg2im and we also increased the types of hyperedges (Miyake et al.,). The trinomial hyperedge allows hsg2im to convolve wider ranges of a scene graph at once, which improves the positional relations of objects. These methods are useful for generating objects with accurate positional relations and thus can often generate natural layouts for scene graphs with a small number of objects. However, they often fail to generate natural layouts for a scene graph with many objects, since these methods use only MLPs for graph convolution, which makes generating object vectors reflecting a wide range of scene graphs difficult. In this paper, we focus on scene-graph-to-image models due to their advantages and aim to overcome their shortcoming, i.e., generating an unnatural layout for a scene graph with many objects. For generating natural layouts for scene graphs with many objects, we propose OA-hsg2im, a new image generation model from a hyper scene graph using object attention.

Some researchers have also attempted to generate natural and high-resolution images from scene graphs. Sortino et al. and Yang et al. have succeeded in generating more realistic images with a VQ-VAE architecture (Van Den Oord et al., 2017) and a diffusion model architecture (Ho et al., 2020), respectively (Sortino et al., 2023; Yang et al., 2022). These models do not employ GAN (Goodfellow et al., 2014)

architecture, and thus can be trained stably to generate realistic and high-resolution images. However, they require a high training cost: the former performs complex processing such as gradient computation and optimization of discrete variables and the latter performs a per-pixel calculation. For example, a diffusion-based image generation model Patch Diffusion (Wang et al., 2023) takes four days for training using 16 V100 GPUs, and a VQVAE-based 2D image to 3D image model PixelSynth consumes about five days for training with four 2080 Ti GPUs (Rockwell et al., 2021). In this paper, we use a pre-trained layout-to-image model LayoutDiffusion (Zheng et al., 2023) as an auxiliary model for generating natural and high-resolution image, which makes us avoid a high training cost.

3 TARGET PROBLEM

3.1 Image Generation from a Hyper Scene Graph

We defined a hyper scene graph as a scene graph with an additional hyperedge which represents a relation among three or more objects (Miyake et al., 2023). Following our previous work, we focus on relations among three objects for simplicity as hyperedges and set our target problem to creating generator $G(H)$ which generates an image $\hat{I}$ from a hyper scene graph $H = (V, E, Q)$. Here, $V = \{v_1, \dots, v_{n_v}\}$ denotes the set of nodes, where n_v represents the number of nodes. E denotes the set of binomial edges in a scene graph, satisfying $E \subseteq V \times \mathcal{R}_2 \times V$, where $\mathcal{R}_2$ is the entire set of labels for binomial relations. Note that for $(v_i, r_j, v_k) \in E$, $i \neq k$. A binomial edge is directed, i.e., (v_i, r_j, v_k) and (v_k, r_j, v_i) are distinct. Q denotes the set of trinomial hyperedges in H , which satisfies $Q \subseteq V \times \mathcal{R}_3 \times V \times V$, where $\mathcal{R}_3$ denotes the entire set of labels for the trinomial relations. A trinomial hyperedge $(v_i, r_j, v_k, v_l) \in Q$ satisfies $i \neq k$, $i \neq l$, $k \neq l$ and is directed as a binomial edge. Figure 2 (b) shows an example of a hyper scene graph.

3.2 Evaluation Metrics

We aim to generate natural layouts and images that are consistent with the input, even for complex scene graphs with multiple objects. We evaluate both the consistency and the naturalness of the output image. First, we discuss the input consistency. Our main objective is to create improved layouts, and we assess the consistency of these layouts by analyzing the

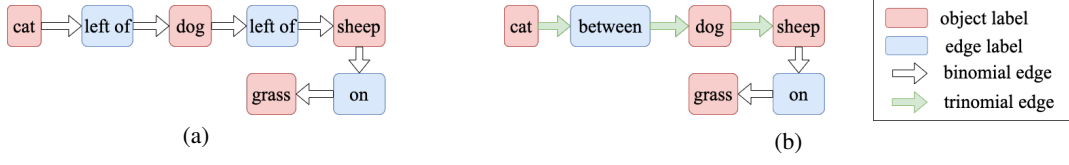


Figure 2: Example of a scene graph (a) and a hyper scene graph (b). This Figure is from (Miyake et al.,). Set V of objects are the same in both (a) and (b), and they are given by $V = \{\text{cat}, \text{dog}, \text{sheep}, \text{grass}\}$. The set of the binomial edges in (a) and (b) are given by $E = \{(v_1, r_1(\text{left of}), v_2), (v_2, r_2(\text{left of}), v_3), (v_3, r_3(\text{on}), v_4)\}$ and $E = \{(v_3, r_3(\text{on}), v_4)\}$, respectively. The set of the trinomial hyperedges in (b) is given by $Q = \{(v_1, r_4(\text{between}), v_2, v_3)\}$. The path $\text{cat} \rightarrow \text{left of} \rightarrow \text{dog}$ corresponds to the binomial edge $(v_1(\text{cat}), \text{left of}, v_2(\text{dog}))$ and represents that a cat is located to the left of a dog. Also, the path $\text{cat} \rightarrow \text{between} \rightarrow \text{dog} \rightarrow \text{sheep}$ corresponds to the trinomial edge $(v_1(\text{cat}), \text{between}, v_2(\text{dog}), v_3(\text{sheep}))$ and represents from left to right, a cat, a dog, and a sheep aligned in a row.

positional relationship between objects connected to the same edge or hyperedge. Intersection over Union (IoU) (Rezatofghi et al., 2019) is one of the evaluation metrics for the object positions in the layouts, which is used in the sg2im paper (Johnson et al., 2018). IoU evaluates the similarity of the layouts between generated and ground truth layouts, which does not assess the consistency between the input and generated layout. The layout which aligns with the input scene graph is not necessarily unique, and our goal is not to generate identical layouts to the ground truth layouts. Therefore, we do not use IoU. We had proposed PTO and AoO as the evaluation metrics for the consistency with the input (Miyake et al., 2023), and then we modified them along with the addition of the types of hyperedge in (Miyake et al.,). In this paper, we use PTO and AoO (Miyake et al.,), which are explained later in Sections 3.2.1 and 3.2.2, respectively.

Second, we explain how to evaluate the naturalness of the generated images. We adopt two metrics, i.e., Inception Score (IS) (Salimans et al., 2016) and Fréchet Inception Distance (FID) (Heusel et al., 2017), which are widely used as the evaluation metrics of the naturalness of the images (Sortino et al., 2023). We explain them in Sections 3.2.3 and 3.2.4, respectively.

3.2.1 Positional Relation of Three Objects (PTO)

PTO (Miyake et al.,) evaluates the proportion of correctly generated the three bounding boxes for each edge type. Let $b_i = (x_{i0}, x_{i1}, y_{i0}, y_{i1})$ be a rectangle bounding box with vertices $(x_{i0}, y_{i0}), (x_{i0}, y_{i1}), (x_{i1}, y_{i0}), (x_{i1}, y_{i1})$, where $x_{i0} < x_{i1}$ and $y_{i0} < y_{i1}$, and $\mathcal{R}_3 = \{\text{between}, \text{stacked}, \text{nested}\}$. A set of bounding boxes $\{b_i, b_j, b_k\}$ corresponding to a hyperedge $(v_i, r_l, v_j, v_k) \in Q$ is considered correctly generated if they satisfy all of the following conditions for each trinomial relation $r \in \mathcal{R}_3$.

The conditions of hyperedge $(v_i, \text{between}, v_j, v_k)$ (Figure 3 (a)) are defined as follows:

1. b_i, b_j, b_k are lined up from left to right in this order without overlapping, i.e., $x_{i0} < x_{i1} < x_{j0} < x_{j1} < x_{k0} < x_{k1}$.
2. b_i, b_j, b_k are not large objects such as the background, i.e., $w_i < 0.7w$ and $w_j < 0.7w$ and $w_k < 0.7w$, where w is the image width and $w_i = x_{i1} - x_{i0}$ is the width of i -th bounding box.
3. b_i, b_j, b_k are of nearly equal size, i.e., $\frac{1}{2} < \frac{w_i}{w_j} < 2$ and $\frac{1}{2} < \frac{h_i}{h_j} < 2$ and $\frac{1}{2} < \frac{w_j}{w_k} < 2$ and $\frac{1}{2} < \frac{h_j}{h_k} < 2$, where $h_i = y_{i1} - y_{i0}$ is the height of the i -th bounding box.
4. b_i, b_j, b_k are not largely apart horizontally, i.e., $0.5 \max(w_i, w_j) > x_{j0} - x_{i1}$ and $0.5 \max(w_j, w_k) > x_{k0} - x_{j1}$.
5. b_i, b_j, b_k are not largely apart vertically, i.e., $0.7 \max(h_i, h_j) > y_{j0} - y_{i1}$ and $0.7 \max(h_j, h_k) > y_{k0} - y_{j1}$.

The conditions of hyperedge $(v_i, \text{stacked}, v_j, v_k)$ (Figure 3 (b)) are the same with the conditions of hyperedge $(v_i, \text{between}, v_j, v_k)$, in which x and y are exchanged. This relation represents the situation that the three bounding boxes of similar sizes are aligned vertically in close positions without overlapping.

The conditions of hyperedge $(v_i, \text{nested}, v_j, v_k)$ (Figure 3 (c)) are defined as follows:

1. b_i, b_j, b_k are not large objects such as the background, i.e., $w_i < 0.7w$ and $w_j < 0.7w$ and $w_k < 0.7w$, where w is the image width and $w_i = x_{i1} - x_{i0}$ is the width of i -th bounding box.
2. The inclusion relation $b_i \supset b_j \supset b_k$ holds horizontally, i.e., $x_{i0} < x_{j0} < x_{k0} < x_{k1} < x_{j1} < x_{i1}$.
3. The inclusion relation $b_i \supset b_j \supset b_k$ holds vertically, i.e., $y_{i0} < y_{j0} < y_{k0} < y_{k1} < y_{j1} < y_{i1}$.

Using the above conditions, PTO for a hyperedge

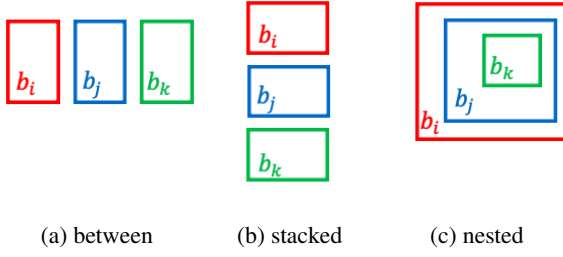


Figure 3: Illustration of the bounding boxes corresponding to the three types of hyperedges (v_i, r_l, v_j, v_k) . This Figure is from (Miyake et al.,).

type r is expressed as follows:

$$\text{PTO}_r = \frac{1}{N_r} \sum_{l=1}^{N_r} J(b_{li}, b_{lj}, b_{lk}), \quad (1)$$

where N_r is the number of the hyperedge r in the test dataset. b_{li}, b_{lj} and b_{lk} are the bounding boxes corresponding to the l -th hyperedge r , and $J(\cdot, \cdot, \cdot)$ is the function which takes 1 if the three bounding boxes satisfy all conditions of hyperedge r and 0 otherwise.

3.2.2 Area of Overlapping (AoO)

AoO (Miyake et al.,) measures the overlap of the three objects connected to a trinomial hyperedge. Note that the objects connected to the hyperedge between and stacked should not overlap each other, while those connected to the hyperedge nested should. AoO is defined differently for the former and the latter trinomial relations as follows:

$$\text{AoO}_r = \begin{cases} \frac{1}{N_r} \sum_{l=1}^{N_r} (\text{IoM}(b_{li}, b_{lj}) + \text{IoM}(b_{li}, b_{lk}) \\ \quad + \text{IoM}(b_{lj}, b_{lk})) & \text{if } r \in \{\text{stacked}, \text{between}\}, \\ 3 - \frac{1}{N_r} \sum_{l=1}^{N_r} (\text{IoM}(b_{li}, b_{lj}) + \text{IoM}(b_{li}, b_{lk}) \\ \quad + \text{IoM}(b_{lj}, b_{lk})) & \text{if } r = \text{nested}, \end{cases} \quad (2)$$

where $\text{IoM}(\cdot, \cdot)$ representing the Intersection over Minimum between two input bounding boxes measures the overlapping of the bounding box as follows:

$$\text{IoM}(X, Y) = \frac{S(X \cap Y)}{\min(S(X), S(Y))}. \quad (3)$$

Here $S(\cdot)$ represents the area of the input region. The smaller AoO is, the better the overlapping of the three objects is controlled.

3.2.3 Inception Score (IS)

We use IS (Salimans et al., 2016) as a measure for evaluating the naturalness of the generated image. IS

is obtained using Inception Network trained on ImageNet (Russakovsky et al., 2015; Szegedy et al., 2015) with the following equation:

$$\text{IS} = \exp(\mathbb{E}_{\hat{I}}[D_{\text{KL}}(p(y|\hat{I})||p(y))]), \quad (4)$$

where $D_{\text{KL}}(\cdot||\cdot)$ is Kullback-Leibler (KL) divergence between distributions. $p(y|\hat{I})$ is the probability distribution of a label y of a given generated image $\hat{I}$ predicted by Inception Network, and $p(y) = \mathbb{E}_{\hat{I}}[p(y|\hat{I})]$ is its marginal probability. A higher IS indicates that the generated images are more natural, as the score increases as the class labels of the generated images become more easily identifiable and more diverse.

3.2.4 Fréchet Inception Distance (FID)

Fréchet Inception Distance (FID) (Heusel et al., 2017), which evaluates the naturalness of the images, is the distance between the distribution of the embedded representations of the ground truth and generated images and is thus consistent with humans' intuition. FID is calculated using Inception Network, the same as IS, with the following equation:

$$\text{FID} = \|m - \hat{m}\|_2^2 + \text{Tr}(C + \hat{C} - 2(C\hat{C})^{1/2}), \quad (5)$$

where m and C are the average vector and the covariance matrix of the feature vectors of the ground truth images obtained from Inception Network, respectively. $\hat{m}$ and $\hat{C}$ are those of the generated images, respectively. A lower FID indicates that the generated images are more natural, as the score decreases when the feature distribution of the generated images is close to that of the ground truth images.

4 ORIGINAL MODEL: hsg2im

hsg2im (Miyake et al., 2023; Miyake et al.,), which is based on an image generation model from a scene graph sg2im (Johnson et al., 2018), generates an image from a hyper scene graph. sg2im has an MLP for the binomial edges net1 and an MLP for the dimension reduction net2 in its graph convolution network. hsg2im has an additional MLP net3 in graph convolutional network of sg2im for processing trinomial hyperedges and uses a pre-trained layout-to-image model Layout2img (He et al., 2021) for generating higher-quality images.

hsg2im generates images as follows: first, objects and edge labels in a hyper scene graph are converted into embedding vectors and inputted into the hyper graph convolutional network. The hyper graph convolutional network converts each object vector so that

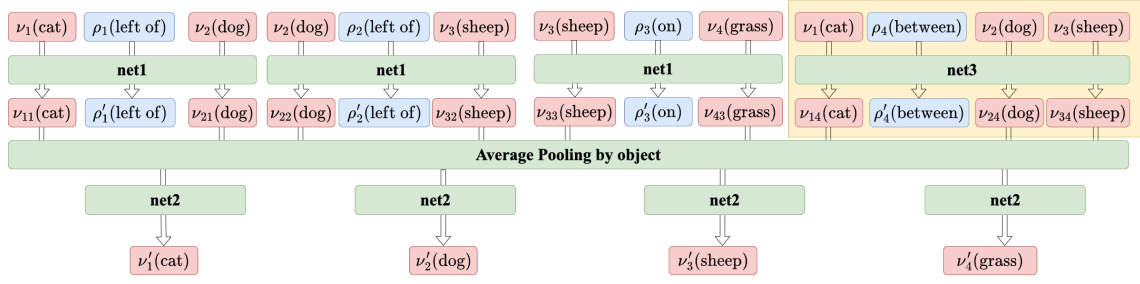


Figure 4: Flowchart of the hyper graph convolutional layer of hsg2im (Miyake et al., 2023; Miyake et al.,). This Figure is from (Miyake et al., 2023). The part without the yellow region corresponds to that of the graph convolutional network of sg2im (Johnson et al., 2018). net1, net2, and net3 represent MLPs for binomial edges, dimension reduction, and trinomial hyperedges, respectively.

it reflects other objects and relations. The graph convolutional network consists of five hyper graph convolutional layers, and the output of the previous layer is used as the input to the next layer. Figure 4 shows the process flow of one of the layers when the scene graph in Figure 2 (b) is the input. The output of the graph are then used by a box regression network to predict bounding boxes. Finally, Layout2img (He et al., 2021) generates an image from the bounding boxes.

Though hsg2im possesses the advantage of easily capturing the relation among three objects, it has also the disadvantage of often failing to generate natural layouts for a hyper scene graph with many objects. The reason is that hsg2im uses only MLPs for graph convolution, which makes generating object vectors reflecting a wide range of scene graphs difficult.

5 PROPOSED MODEL: OA-hsg2im

We propose Object Attention hsg2im (OA-hsg2im) to address the shortcoming of hsg2im, i.e., generating an unnatural layout for a scene graph with many objects. An overview of the generator of OA-hsg2im is shown in Figure 5. The modifications in OA-hsg2im from hsg2im are summarized as follows:

- OA-hsg2im introduces object attention layers to reflect the relation between distant objects.
- OA-hsg2im performs a positional encoding for object vectors to give the positional information in the hyper scene graph object vectors.
- OA-hsg2im employs a pre-trained layout-to-image model LayoutDiffusion (Zheng et al., 2023) to generate higher-quality images than Layout2img (He et al., 2021).

In this Section, we explain these three modifications.

OA-hsg2im has object attention layers before each of the graph convolutional layers. There are N_l object

attention layers and N_l graph convolution layers and the output of the previous layer is used as the input of the next layer. A flow of the i -th object attention layer and hyper graph convolutional layer is shown in Figure 6. The i -th object attention layer converts object vectors D_{i-1} using attention scores, which are computed for all combinations of two objects in a hyper scene graph, with Multi-Head Attention, which is the same architecture with self-attention in transformer (Vaswani et al., 2017). The attention of the j -th head in the i -th object attention layer is performed as follows:

$$\text{Attention}(\text{Query}, \text{Key}, \text{Value}) = \text{softmax} \left(\frac{\text{Query} * \text{Key}^\top}{\sqrt{d/N_h}} \right) \text{Value}, \quad (6)$$

where $\text{Query} = D_{i-1}W_{ij}^Q$, $\text{Key} = D_{i-1}W_{ij}^K$ and $\text{Value} = D_{i-1}W_{ij}^V$. N_h is the number of the heads. The conversions from D_{i-1} to Query , Key , Value are performed with the linear layers and $W_{i,j}^Q$, $W_{i,j}^K$, $W_{i,j}^V$ are the weight of these layers. The vectors obtained from each head are concatenated and fed into a linear layer. The resulting object vectors and edge features calculated from relation vectors F_{i-1} are fed into hyper graph convolutional layer, which is used in the hsg2im (Miyake et al., 2023; Miyake et al.,). We obtain object vectors $D_i \in \mathbb{R}^{n_v \times d}$ and relation vectors $F_i \in \mathbb{R}^{n_r \times d}$, where n_v is the number of objects, n_r is the add-sum of the number of edges and hyperedges and d is the dimension of object and relation vectors.

We gain D_0 and F_0 from the object embedding network and the relation embedding network, respectively. The attention score, which represents the degree of the relevance of two objects, makes the object attend to other ones regardless of the distance between them in the hyper scene graph. Therefore, we can obtain object vectors reflecting a wider range of objects in the hyper scene graph than hsg2im (Miyake et al., 2023; Miyake et al.,) and thus OA-hsg2im is

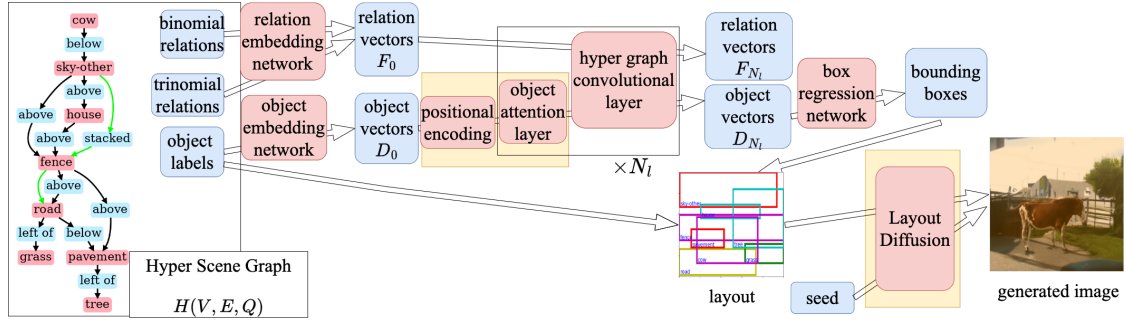


Figure 5: Example of generating an image with OA-hsg2im. The modified parts from hsg2im (Miyake et al.,) are highlighted in yellow. $N_l (= 12)$ is the number of object attention layers and graph convolutional layers.

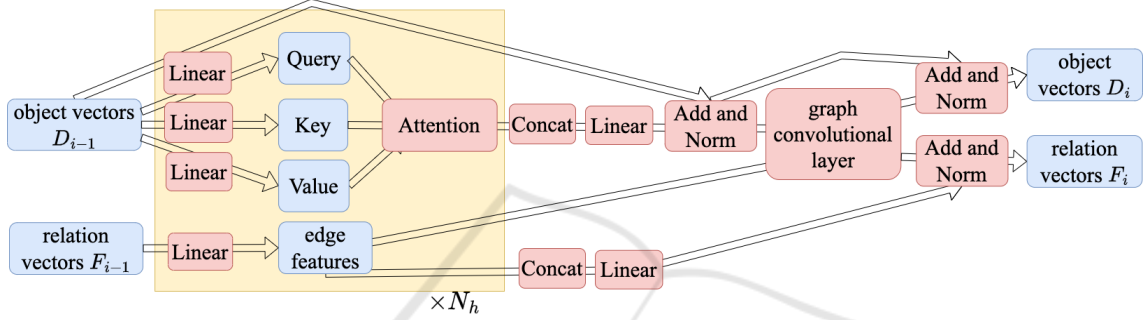


Figure 6: Flow of the i -th object attention layer and hyper graph convolutional layer. In the Add and Norm, we perform the batch normalization to the add-sum of the two input vectors. In the Concat, we concatenate the object vectors obtained from each head of attention.

expected to generate more natural layouts.

Positional encoding is conducted to give the positional information in the scene graph object vectors. We use graph Laplacian as the positional encoding following the graph transformer (Dwivedi and Bresson, 2020). The graph Laplacian $\Delta \in \mathbb{R}^{n_v \times n_v}$ are calculated as follows:

$$\Delta = I - M^{-1/2} A M^{-1/2}, \quad (7)$$

where $I \in \mathbb{R}^{n_v \times n_v}$ is the identity matrix, $M \in \mathbb{R}^{n_v \times n_v}$ and $A \in \mathbb{R}^{n_v \times n_v}$ represent the degree matrix and the adjacency matrix, respectively. As in the graph transformer paper (Dwivedi and Bresson, 2020), we add positional encoding vectors $T \in \mathbb{R}^{n_v \times d}$, which are converted from $\Delta \in \mathbb{R}^{n_v \times d}$ with a linear layer, to object vectors D_0 .

Next, we describe image generation using LayoutDiffusion (Zheng et al., 2023). To generate higher-resolution images (256×256 pixels), we use a pre-trained layout-to-image model LayoutDiffusion as an auxiliary model. LayoutDiffusion takes layout $L = \{(v_i, b_i)\}_{i=1}^{n_v}$ as input. Set $V = \{(v_i)\}_{i=1}^{n_v}$ of objects is obtained from the input hyper scene graph and set $\hat{B} = \{(b_i)\}_{i=1}^{n_v}$ of bounding boxes is generated by OA-hsg2im. LayoutDiffusion has a diffusion architecture (Ho et al., 2020) and thus can generate higher-quality

and higher-resolution images than GAN-based methods.

6 PREJUDICE IN LayoutDiffusion

We investigate prejudice about genders in pre-trained LayoutDiffusion (Zheng et al., 2023) as follows. We generate a layout for each of the three hyper scene graphs in Figure 8 with OA-hsg2im trained on the VG dataset (Krishna et al., 2017), which is explained in Section 7.1. We generate ten images, each has three persons, for each of the three layouts in Figure 8 by changing the seed of the random function with LayoutDiffusion (Zheng et al., 2023) pre-trained on the VG dataset. We examine how the hyper scene graphs affect the gender of the persons in the generated images by counting the number of males and females. Examples and the results are shown in Figure 8 and Table 1, respectively.

Hyper scene graph in (a) has no object with gender bias; however, 19 males were generated, while only 10 females were generated. These results indicate that LayoutDiffusion has learned a prejudice in VG dataset (Krishna et al., 2017), in which males are more common than females as persons. In example

Table 1: Results of counting the males and females for person in the 10 generated images for each of the six hyper scene graphs. If the image was too dark or blurred to determine the gender, it is assumed to be indistinguishable. Examples (a)-(c) correspond to Examples (a)’-(c)’, respectively. Each hyper scene graph of the formers has three persons while all of persons are replaced with women in the hyper scene graphs of the latters.

	male	female	indistinguishable
Example (a)	19	10	1
Example (b)	25	2	3
Example (c)	30	0	0
Example (a)’	0	27	3
Example (b)’	3	27	0
Example (c)’	4	26	0

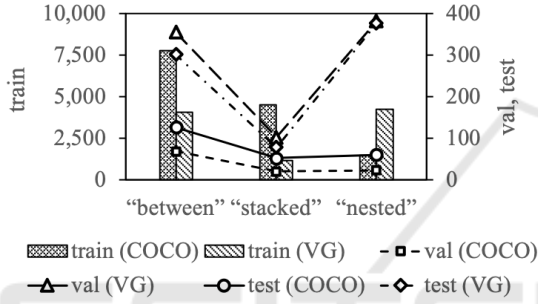


Figure 7: Numbers of three types of trinomial relations in each dataset.

(b), we specify that three persons are wearing tie in the generated image and obtained 25 males and 2 females. In the same way, we specify that three persons are wearing glove in example (c) in the generated image. We obtained 30 males and 0 female. These results indicate that LayoutDiffusion has learned a prejudice in VG dataset (Krishna et al., 2017), in which tie and glove are more common for males than females.

Though OA-hsg2im does not handle this gender bias, the addition of the woman images to the training sets will alleviate this bias. Also, we confirmed that the images of the woman are properly generated with the change of the object label from person to woman in Example (a)’-(c)’ in Figure 8 and Table 1.

7 EXPERIMENTS

7.1 Dataset

We use the COCO Stuff (COCO) (Caesar et al., 2018) and Visual Genome (VG) (Krishna et al., 2017) datasets with additional trinomial hyperedges as in previous works (Johnson et al., 2018; Miyake et al.,

2023; Miyake et al.,). These datasets are the same as the our recent work (Miyake et al.,), except for the test sets. In this paper, we use only samples that include hyperedges as test sets.

The COCO (Caesar et al., 2018) consists of images, objects, their bounding boxes, and segmentation masks. Since the COCO dataset does not contain data on relations between objects, we construct a hyper scene graph by adding nine relations left of, right of, above, below, inside, surrounding between, stacked, and nested based on the bounding boxes. Based on the conditions in Section 3.2.1, the three types of relations, i.e., between, stacked and nested, are added as trinomial hyperedges in hsg2im. The conditions in adding edges $(v_i, \text{left of}, v_j)$ and $(v_j, \text{right of}, v_i)$ are the same as hyperedge $(v_i, \text{between}, v_j, v_k)$. Similarly, the conditions in adding edges (v_i, above, v_j) and (v_j, below, v_i) are the same as hyperedge $(v_i, \text{stacked}, v_j, v_k)$ in which v_k is ignored, and the conditions in adding edges $(v_i, \text{surrounding}, v_j)$ and $(v_j, \text{inside}, v_i)$ are the same as hyperedge $(v_i, \text{nested}, v_j, v_k)$. The number of these types of relations are shown in Figure 7. There are 40,000 training images and 5,000 validation images. Since the COCO dataset has no test set, we divide the validation set into a new validation set and a test set and exclude the samples whose scene does not have a hyperedges from the test set. As a result, we obtain 24,972 training, 1,667 validation, and 131 test images.

The VG (Krishna et al., 2017) version 1.4 dataset, containing 108,077 images annotated with scene graphs, consists of images, objects, their bounding boxes, and the binomial relations between the objects. In the entire dataset, we use objects and relations which appear more than 2,000 and 500 times, respectively, following (Johnson et al., 2018). Samples with less than 2 objects and more than 31 objects were ignored. We exclude the samples whose scene does not have a hyperedge from the test set as in the COCO. As the result, we use 62,602 training, 5,069 validation, and 755 test images. For the VG, we add the trinomial relations as in the COCO. The number of these types of relations are shown in Figure 7.

7.2 Implementation Details

We train OA-hsg2im on each of the COCO and Visual Genome datasets, respectively. We use the optimizer Adam (Kingma and Ba, 2015) with the epoch upper bound $(= 3 \times 10^2)$. We use the number of attention layer $N_l = 12$ and the attention head $N_h = 12$, and set the dimension of object and relation vectors $d = 768$.

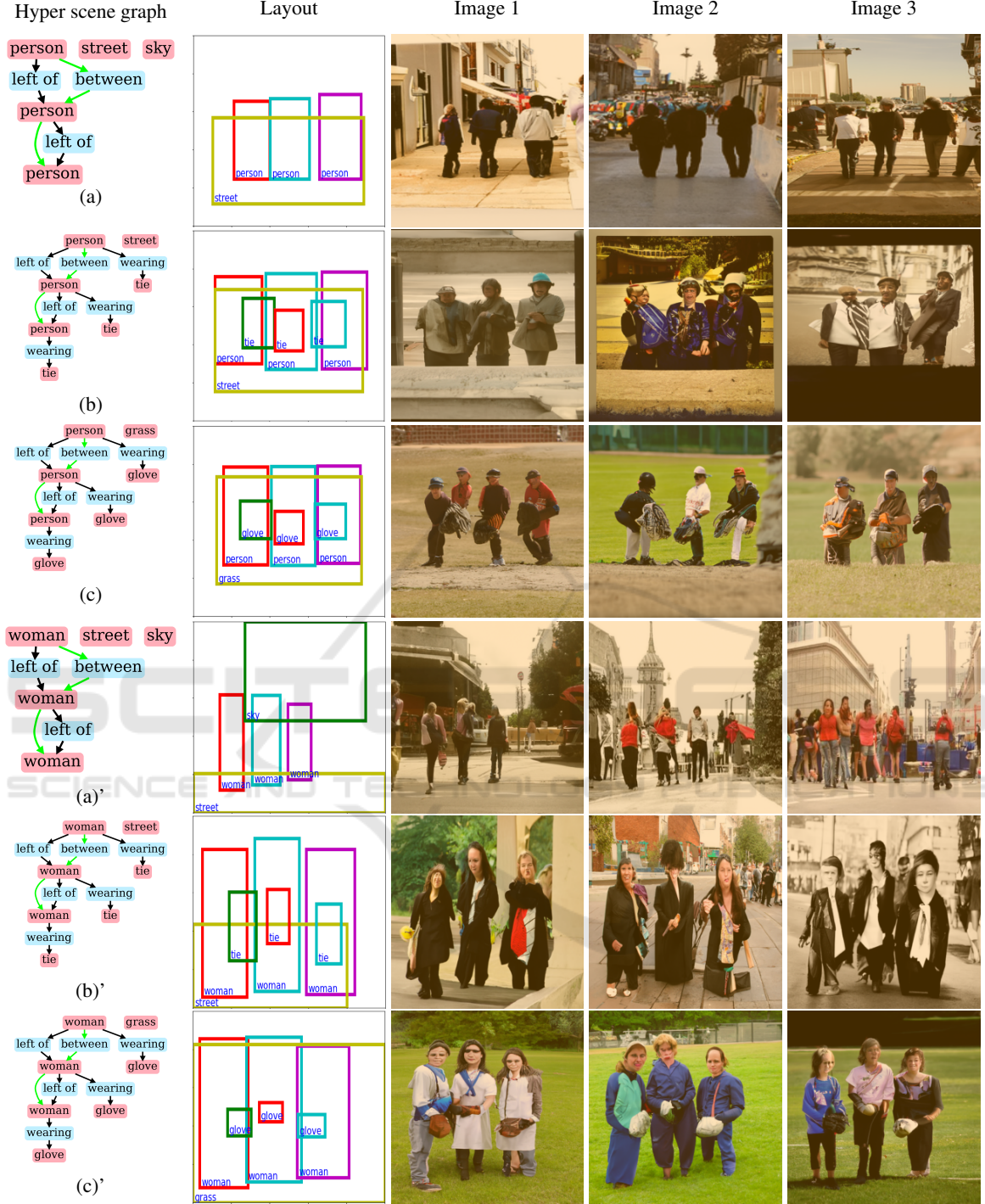


Figure 8: Examples of hyper scene graphs, layouts, and images in the investigation of prejudice about genders in LayoutDiffusion (Zheng et al., 2023). We show three generated images for each hyper scene graph. Examples (a)-(c) correspond to Examples (a')-(c'), respectively. Each hyper scene graph of the formers has three persons while all of persons are replaced with women in the hyper scene graphs of the latters.

We use the loss function L defined as follows:

$$L = w_{\text{MSE}} * \text{MSE}(B, \hat{B}) + w_{\text{RBL1}} * L_{\text{RBL1}} + w_{\text{RBL2}} * L_{\text{RBL2}}, \quad (8)$$

where MSE means Mean Squared Error and $w_{\cdot}$ represents the weight of each loss. B and $\hat{B}$ represent a set of ground truth and generated bounding bounding

boxes, respectively. We use $w_{\text{MSE}} = 1$, $w_{\text{RBL1}} = 1$, $w_{\text{RBL2}} = 2 * 10^{-3}$. L_{RBL1} and L_{RBL2} are the losses in terms of the relative positions between the two objects connected to the same edge and hyperedge proposed in our recent work (Miyake et al.,), respectively.

The former penalizes the difference of relative vectors, which represent the relative position of the two objects connected to a binomial edge, between the ground truth and the generated bounding boxes. The latter penalizes the difference of binary vectors, which represent larger and smaller relations of spatial coordinates of the two objects.

Relative box loss function L_{RBL1} is defined as follows:

$$L_{\text{RBL1}} = \frac{1}{N} \sum_{i=1}^N \text{MSE}(d_i, \hat{d}_i), \quad (9)$$

where N is the number of all types of binomial edges in one batch, and d_i and $\hat{d}_i$ are the relative vectors of the ground truth and generated bounding boxes, respectively. Here, d_i is defined by $d_i = b_{is} - b_{io}$, where b_{is} and b_{io} are bounding boxes corresponding to the initial and terminate vertices of i -th binomial edge, respectively. For example, when the i -th edge is (cow, below, sky-other) in the hyper scene graph in Figure 5, $b_{is} = b_{\text{cow}}$ and $b_{io} = b_{\text{sky-other}}$.

Relative box loss function L_{RBL2} is defined as follows:

$$L_{\text{RBL2}} = \frac{1}{N} \sum_{i=1}^N \text{BCE}(g_i, \hat{g}_i), \quad (10)$$

where BCE means the Binary Cross Entropy, and g_i and $\hat{g}_i$ represent the 8-dimensional binary vectors of the ground truth and generated bounding boxes, respectively. Here, g_i is defined as $g_i = (\mathbb{I}(x_{si0} < x_{oi0}), \mathbb{I}(x_{si1} < x_{oi0}), \mathbb{I}(x_{si0} < x_{oi1}), \mathbb{I}(x_{si1} < x_{oi1}), \mathbb{I}(y_{si0} < y_{oi0}), \mathbb{I}(y_{si1} < y_{oi0}), \mathbb{I}(y_{si0} < y_{oi1}), \mathbb{I}(y_{si1} < y_{oi1}))$ and $\hat{g}_i$ is defined similarly, where $\mathbb{I}(\cdot)$ is an indicator function which returns 1 when the input condition holds and 0 otherwise.

7.3 Results and Discussion

We evaluate OA-hsg2im by comparing it with hsg2im (Miyake et al.,) and hsg2im*. hsg2im* is the model trained with the same conditions of OA-hsg2im except for the object attention layers. These models use the same loss function as Eq. (8). In order to align the conditions with OA-hsg2im, we use LayoutDiffusion (Zheng et al., 2023) instead of Layout2img (He et al., 2021) as a pre-trained layout-to-image model in hsg2im. The quantitative results on the COCO and the VG are shown in Table 2 (a) and (b), respectively. We can see that OA-hsg2im shows high PTO scores

more consistently than the other models and all models show low AoO scores in both datasets. Also, PTO and AoO scores tend to be better on the COCO than the VG. The reason would lie in the number of the types of the relations, i.e., the COCO has nine types of relations and the VG has 48 types of relations. If the number of the types is smaller, models can capture the relations more easily.

In terms of IS, hsg2im* and hsg2im perform better than OA-hsg2im on the COCO and the VG, respectively. We obtain two FID for each model, i.e., the distance between the sets of the ground truth image and the generated images of each model and the distance between the sets of the generated images from the ground truth layout and each model. In terms of the former FID, OA-hsg2im shows better scores than hsg2im* on both datasets. As for the latter FID, the images generated from the same hyper scene graph have the same atmospheres because the seed values are fixed. In terms of the latter FID, OA-hsg2im shows better scores on the COCO dataset and slightly worse scores on the VG dataset. These results show that OA-hsg2im generates more natural and consistent layouts and images.

Next, we qualitatively evaluate the generated images. Figures 9, 10 show the generated images of hsg2im (Miyake et al.,) and OA-hsg2im, on the COCO and the VG datasets, respectively. In the example (b) in Figure 9, the hyper scene graph has ten objects. In the layout generated by hsg2im (Miyake et al.,), the objects are not arranged naturally and consistently. In the example of (b), many objects, i.e., floor-wood (red box), table (purple) and spoon (purple), and bowl (red), overlap each other in the middle center, and they cannot be seen in the generated image of hsg2im. On the other hand, floor-wood (red), table (purple), spoon (purple) are arranged naturally in the layout generated by OA-hsg2im, and two persons do not overlap each other. In the example of (f) in Figure 10, the hyper scene graph has a hyper-edge woman (light blue) - between - woman (purple) $\rightarrow$ person (red); however, woman (light blue) and woman (purple) overlap each other in the layout generated by hsg2im, and we cannot see woman (light blue) clearly. Conversely, they do not overlap at all in the layout generated by OA-hsg2im, and we can see woman (light blue). Thus, we can also confirm the effectiveness of OA-hsg2im in generating natural and consistent layouts and images.

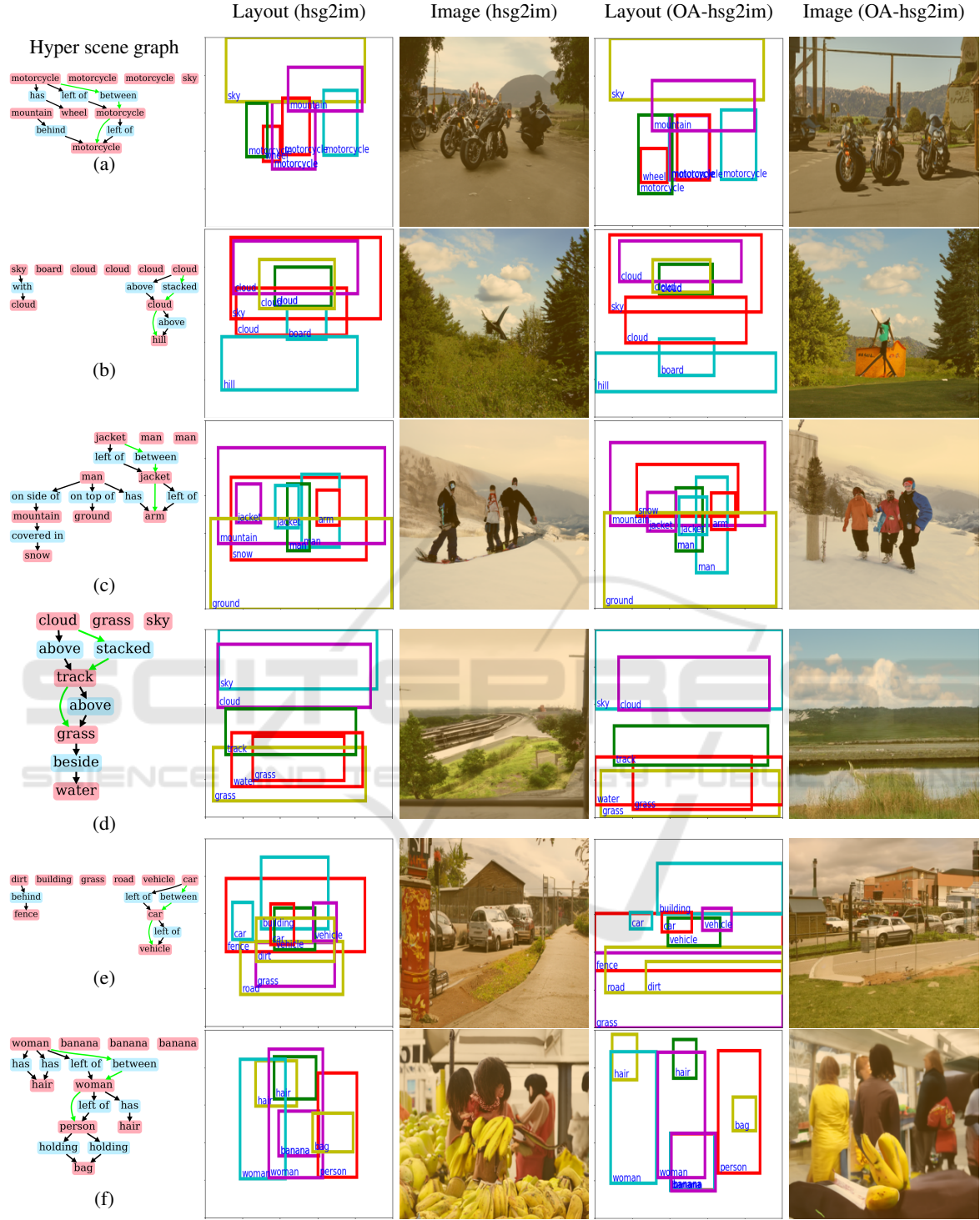


Figure 10: Comparison of images generated by hsg2im (Miyake et al.,) and OA-hsg2im on the VG datasets.

erate more high-resolution images. The object attention layers convert object vectors so that they attend to other objects relevant to themselves, regardless of the distance between them in a scene graph, which

allows OA-hsg2im generate more natural and consistent images with the input. Therefore, OA-hsg2im can alleviate the problem of hsg2im, i.e., the unnatural layouts for the complex situations, which is our

Table 2: Quantitative results on (a) the COCO dataset (Caesar et al., 2018) and (b) the VG dataset (Krishna et al., 2017). In FID, the values in parentheses are the distance between the sets of images generated from G.T. (grand truth) Layout and each model. In G.T. Layout, we obtain IS and FID for the generated images generated from LayoutDiffusion using ground truth layouts. In G.T. Image, we obtain IS for the ground truth images.

(a)								
		PTO $\uparrow$			AoO $\downarrow$			
	between	stacked	nested	between	stacked	nested	IS $\uparrow$	FID $\downarrow$
hsg2im	0.89	0.63	0.77	0.004	0.005	0.09	14.14	196.81 (166.0)
hsg2im*	0.87	0.58	0.90	0.003	0.017	0.003	14.89	187.5 (156.7)
OA-hsg2im	0.82	0.79	0.97	0.004	0.002	0.006	14.53	186.1 (151.84)
G.T. Layout	-	-	-	-	-	-	14.23	180.0
G.T. Image	-	-	-	-	-	-	17.76	-
(b)								
		PTO $\uparrow$			AoO $\downarrow$			
	between	stacked	nested	between	stacked	nested	IS $\uparrow$	FID $\downarrow$
hsg2im	0.50	0.44	0.86	0.001	0.042	0.04	20.89	84.62 (66.32)
hsg2im*	0.78	0.65	0.89	0.009	0.004	0.005	18.41	80.83 (64.16)
OA-hsg2im	0.73	0.71	0.93	0.007	0.008	0.003	19.34	76.47 (68.12)
G.T. Layout	-	-	-	-	-	-	21.05	76.04
G.T. Image	-	-	-	-	-	-	25.11	-

main contribution. The results in terms of PTO and FID show that OA-hsg2im has succeeded in improving the consistency and naturalness of the generated image.

However, we see that the generated images are less natural than the ground truth images. Transformer (Vaswani et al., 2017) has two types of attentions, i.e. the self-attention, which is employed in OA-hsg2im, and the source-target attention. The latter attention uses different vectors for obtaining Key and Query, and is thus useful for combining two feature spaces, e.g., English and Spanish feature spaces, in the natural language translation tasks. We can use this attention in the image generation from hyper scene graphs for combining the object vector space and the object bounding box spaces, which would make the models learn the complex positional relation between objects and improve the naturalness of the generated layouts. We believe that this is an interesting direction for our future work.

ACKNOWLEDGEMENTS

A part of this work was supported by JSPS KAKENHI Grant Number JP21K19795.

REFERENCES

Caesar, H., Uijlings, J., and Ferrari, V. (2018). Coco-Stuff: Thing and Stuff Classes in Context. In *Proceedings of*

the IEEE Conference on Computer Vision and Pattern Recognition, pages 1209–1218.

Dwivedi, V. P. and Bresson, X. (2020). A Generalization of Transformer Networks to Graphs. *ArXiv*, abs/2012.09699.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 27.

He, S., Liao, W., Yang, M. Y., Yang, Y., Song, Y.-Z., Rosenhahn, B., and Xiang, T. (2021). Context-Aware Layout to Image Generation with Enhanced Object Appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15049–15058.

Herzig, R., Bar, A., Xu, H., Chechik, G., Darrell, T., and Globerson, A. (2020). Learning Canonical Representations for Scene Graph to Image Generation. In *European Conference on Computer Vision*.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Advances in Neural Information Processing Systems*.

Hinz, T., Heinrich, S., and Wermter, S. (2022). Semantic Object Accuracy for Generative Text-to-Image Synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44:1552–1565.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising Diffusion Probabilistic Models. *Advances in Neural Information Processing Systems*, 33:6840–6851.

Johnson, J., Gupta, A., and Fei-Fei, L. (2018). Image Generation from Scene Graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1219–1228.

- Kingma, D. P. and Ba, J. (2015). Adam: A Method for Stochastic Optimization. *Proceedings of the International Conference on Learning Representations*.
- Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., et al. (2017). Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision*, 123(1):32–73.
- Miyake, R., Matsukawa, T., and Suzuki, E. Image Generation from Hyper Scene Graph with Multi-Types of Trinomial Hyperedges. *special issue of the Springer Nature Computer Science Journal (under submission)*.
- Miyake, R., Matsukawa, T., and Suzuki, E. (2023). Image Generation from a Hyper Scene Graph with Trinomial Hyperedges. In *Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISI-GRAPP 2023) - Volume 5: VISAPP*, pages 185–195.
- Odena, A., Olah, C., and Shlens, J. (2017). Conditional Image Synthesis with Auxiliary Classifier GANs. In *Proceedings of the International Conference on Machine Learning*, pages 2642–2651. PMLR.
- Reed, S. E., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. (2016). Generative Adversarial Text to Image Synthesis. *ArXiv*, abs/1605.05396.
- Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., and Savarese, S. (2019). Generalized Intersection over Union: A Metric and a Loss for Bounding Box Regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 658–666.
- Rockwell, C., Fouhey, D. F., and Johnson, J. (2021). Pixel-synth: Generating a 3D-Consistent Experience from a Single Image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14104–14113.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252.
- Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. (2016). Improved Techniques for Training GANs. *Advances in Neural Information Processing Systems*, 29.
- Schuster, S., Krishna, R., Chang, A., Fei-Fei, L., and Manning, C. D. (2015). Generating Semantically Precise Scene Graphs from Textual Descriptions for Improved Image Retrieval. In *Proceedings of the Fourth Workshop on Vision and Language*, pages 70–80.
- Sortino, R., Palazzo, S., and Spampinato, C. (2023). Transformer-based Image Generation from Scene Graphs. *Computer Vision and Image Understanding*, vo.233.
- Sun, W. and Wu, T. (2019). Image Synthesis From Reconfigurable Layout and Style. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10531–10540.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going Deeper with Convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9.
- Tang, R., Du, M., Li, Y., Liu, Z., Zou, N., and Hu, X. (2021). Mitigating Gender Bias in Captioning Systems. In *Proceedings of the Web Conference 2021*, pages 633–645.
- Van Den Oord, A., Vinyals, O., et al. (2017). Neural Discrete Representation Learning. *Advances in Neural Information Processing Systems*, 30.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is All You Need. *Advances in Neural Information Processing Systems*, 30.
- Vo, D. M. and Sugimoto, A. (2020). Visual-Relation Conscious Image Generation from Structured-Text. In *Proceedings of European Conference on Computer Vision*.
- Wang, Z., Jiang, Y., Zheng, H., Wang, P., He, P., Wang, Z., Chen, W., and Zhou, M. (2023). Patch diffusion: Faster and More Data-Efficient Training of Diffusion Models. *arXiv preprint arXiv:2304.12526*.
- Yang, L., Huang, Z., Song, Y., Hong, S., Li, G., Zhang, W., Cui, B., Ghanem, B., and Yang, M.-H. (2022). Diffusion-Based Scene Graph to Image Generation with Masked Contrastive Pre-Training. *ArXiv*, abs/2211.11138.
- Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. N. (2017). StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5907–5915.
- Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. N. (2018). Stackgan++: Realistic Image Synthesis with Stacked Generative Adversarial Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1947–1962.
- Zhang, K., Shinden, H., Mutsuro, T., and Suzuki, E. (2022). Judging Instinct Exploitation in Statistical Data Explanations Based on Word Embedding. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, pages 867–879.
- Zheng, G., Zhou, X., Li, X., Qi, Z., Shan, Y., and Li, X. (2023). LayoutDiffusion: Controllable Diffusion Model for Layout-to-Image Generation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22490–22499.