Label2im: Knowledge Graph Guided Image Generation from Labels

Hewen Xiao[1]
hellomimimi@mail.dlut.edu.cn
Yuqiu Kong*[1]
yqkong@dlut.edu.cn
Hongchen Tan[2]
tanhongchenphd@bjut.edu.cn
Xiuping Liu[1]
xpliu@dlut.edu.cn
Baocai Yin[1]
ybc@dlut.edu.cn

[1] Dalian University of Technology, Dalian, China
[2] Beijing University of Technology, Beijing, China

Abstract

Most recent generation methods synthesize images from either complex textual descriptions or scene graphs. However, users need to elaborate attributes and relationships of objects in the scene, and scene graphs are more difficult to obtain. To simplify the burden of users, in this work, we propose a Label2im model to generate images from object labels directly with the help of a Knowledge Graph (KG), *e.g*. Visual Genome. To acquire rational interactions between objects, we explore possible relationships from the KG. Considering that there is a large gap between the label domain and image domain, we propose to learn knowledge representations of the scene graph from the KG to ensure the semantic consistency. First, given several object labels, we design a Scene Graph Selection Module (SGSM) to explore interactions between objects in the KG and generate a set of scene graphs. Second, the structure representation and knowledge embedding of the scene graph are learned and integrated in the Scene Graph Representation Module (SGRM), which leads to rational scene layouts. Based on the scene layouts and KG, we employ the Cascaded Refinement Network (CRN) to generate the final image. To encode knowledge information in the generation process, we propose a Triplet Attention Module (TAM) which is embedded in the CRN. We verify the effectiveness of the proposed method on the Visual Genome dataset and demonstrate that our method is able to generate complex images with rich content and fine details.

1 Introduction

Complex scene generation provides great potential for artistic creation, especially for non-professionals. Users specifying their demands, scene generation models synthesize images respecting the constraints. For most existing generation methods, users need to provide detailed textual descriptions of the appearance of objects and their interactions [18, 30] or elaborate scene graphs which indicate the structure of the scene [1, 11, 14, 19], or design

* indicates the corresponding authour.

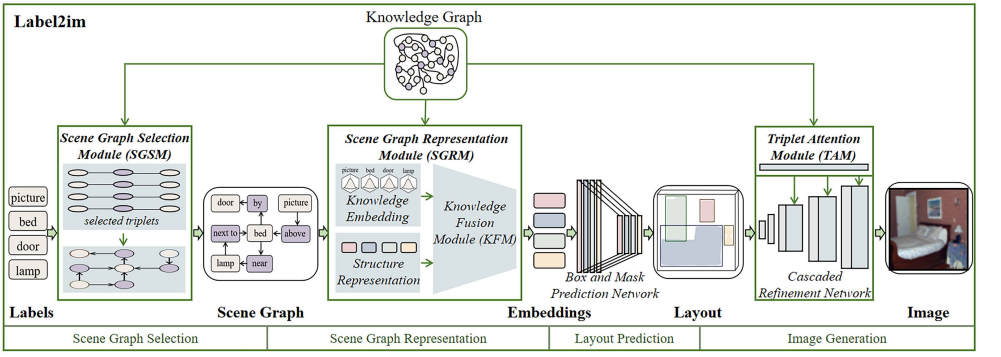


Figure 1: The overall architecture of the proposed method, which consists of scene graph selection, scene graph representation, layout prediction, and image generation.

layouts which specify the spatial relationship of objects [29, 42]. However, all these inputs increase the burdens of users and limit the model's creativity. To this end, in this work, we consider generating complex images directly from object labels.

Because of the domain shift problem between multi-modality data, it is non-trivial to synthesize images from the semantic domain. In the text-to-image generation methods [18, 30, 36, 40], the uncertainty of the description leads to an uncontrollable generation process. Therefore, it is necessary to provide specific instructions, such as the attributes and locations of objects and the interactions among them, which raises demands for users and requires generation modules to have a high-level understanding of textual descriptions and visual concepts. In comparison, scene graphs [13] consist of objects as nodes and relationships as edges and encode the structure information of the scene, which contributes to a better understanding of the configuration of objects in the image. However, scene graphs are difficult to design, thus making this application hard to be used prevalently. Some works [29, 42] learn to generate images from scene layouts which contain specified category and position of each object, but the layout constraints limit the structural diversity of generated images.

Compared with the above mentioned tasks, synthesizing realistic images directly from object labels is more user-friendly. However, it is challenging to project object labels to the image domain for two reasons. First, label-to-image generation is a one-to-many mapping problem, in which the lacking of attribute and relationship information of objects may lead to unreasonable appearance and arrangement of objects in the generated images. Second, labels only provide basic conceptual information, while images incorporate rich semantic content. Therefore, there is a large gap between object labels and image domain, which requires sufficient information and cross-domain semantic consistency.

Our method, Label2im, exploits the KG to alleviate the above issues. For the first issue, we first construct a KG from the Visual Genome [17]. The KG consists of extensive triplets indicating the attributes and relationships of common-seen objects in realistic scenes, that build a bridge between the semantic domain and image domain. Given object labels, we randomly select triplets which contain the objects from the KG and form a scene graph. For the second issue, considering that the KG incorporates a large amount of object categories and their interactions, and besides the specified objects, knowledge embeddings of the scene graph can introduce external information and lead to rational scene layouts, we construct a Scene Graph Representation Module (SGRM) to learn knowledge embeddings of the scene graph from the KG, and then combine the structure information and knowledge embeddings

of the scene graph by a Knowledge Fusion Module (KFM). In addition, in the image generation stage, we propose a Triplet Attention Module (TAM) to enforce the in-depth integration of visual feature and knowledge representation. The overall Label2im network involves four cascaded stages: first, SGSM explores relationships of objects in the KG and forms scene graphs; second, SGRM learns structure information and knowledge embeddings of the scene graph and combines them by the KFM; third, a scene layout is predicted based on scene graph representation; a final image is generated by a CRN with TAMs considering both visual and knowledge information.

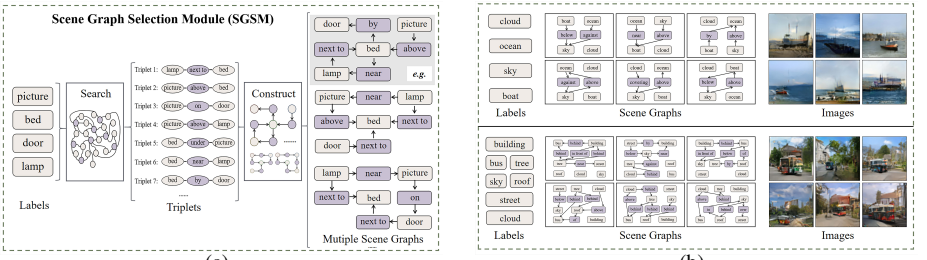
Our main contributions are three folds: (1) A Label2im network which directly synthesizes images from object labels by the help of the KG. (2) SGSM which automatically selects scene graph from the KG giving object labels; SGRM represents scene graphs by structure information and knowledge embeddings; TAM introduces knowledge representations to ensure semantic consistency in the image generation process; (3) Quantitative and qualitative experiments on the Visual Genome demonstrate that the proposed method is able to generate realistic images with fine details.

2 Related Work

Conditional Image Generation. Conditional image generation aims to synthesize images respecting the constraints of inputs. The researches of autoregressive approaches [32], Variational Autoencoders [16], and Generative Adversarial Networks (GAN) [7] lay the foundation for the development of conditional image generation. The work [20] first proposes the conditional GAN which generates images given class labels. Based on this architecture, various single-image generated methods are proposed [5, 6, 22]. In recent works, researchers employ different kinds of conditions to represent the scene aiming to generate semantically complex images, such as textual descriptions [10, 18, 30, 37], scene graphs [1, 14, 19, 31], and layouts [3, 23, 26, 29, 42]. Specifically, Chen and Koltun [3] propose a CRN to generate high-quality images conditioned on pixel-wise semantic layouts. Layout2im [42] generates images from scene layouts based on disentangled representations and learns appearance information of objects from image crops. Compared to most text-to-image generation methods [27, 36, 39] that focus on the flower [21] and bird datasets [33], Obj-GAN [18] outputs images with complicated scenes from textual descriptions and refines image details with an object-driven attention mechanism. However, the linear structure of the text feeds the model with redundant information.

The works of image generation from scene graphs are more related to our work. Given a scene graph, sg2im [14] utilizes Graph Convolution Network (GCN) to encode the scene graph, then predicts bounding boxes and masks of objects to form a scene layout, and finally employ the CRN to generate the images. Inspired by sg2im [14], PasteGAN [19] proposes a semi-parametric method that additionally inputs optional image crops to improve the image quality. Considering that it may be inconvenient for users to design layouts, textual descriptions, or scene graphs, our method aims to synthesize realistic images directly from object labels.

Knowledge Graph and Knowledge Representation. KG [4, 24] is a semantic graph that stores relation facts, *e.g.* the triplet $(head\,entity, relationship, tail\,entity)$. The emergence of it has expanded the research ideas for computer vision tasks. To enforce semantic consistency during scene parsing, KE-GAN [25] designs an extra knowledge relation loss and employs random walk to capture semantic consistencies between labels from KG. Moreover, Zareian *et al*. [38] utilize a graph-based neural network to bridge the mapping between KG



(a) (b)

Figure 2: (a) SGSM. Given the object labels and the KG, scene graphs can be constructed by randomly selecting triplets from the KG. (b) Scene graphs constructed by the SGSM and the corresponding generated images.

and scene graphs for better predicting the scene graph from a given image. In this work, we are concerned with the role of KG for scene generation.

Knowledge representation aims to learn feature embeddings of entities and relationships in KG. The translation-based knowledge representation methods [2, 8, 12, 34, 35] adopt an energy-based framework and obey the translation principle $h-t \approx r$, where h,r,t indicate the representations of *head entity*, *relationship*, and *tail entity* in the triplet, respectively. In this work, we employ KG2E [8] to extract knowledge representations of objects and relationships and introduce the external information in the generation process.

3 Proposed Method

3.1 Overview

The overall architecture of the proposed method is illustrated in Figure 1. Given the KG and a set of object labels, the proposed Label2im generates realistic images. First, SGSM randomly selects a set of triplets which incorporates the given objects in the KG to form scene graphs. Second, SGRM learns the representation of scene graph. In SGRM, a GCN is employed to encode the structure information of the scene graph; a knowledge embedding algorithm, KG2E [8] is applied to learn knowledge representations of objects and relationships in the KG; and then the structure information and knowledge embeddings are combined by the KFM. Then, a scene layout is output based on the fused features by predicting the bounding box and mask of each object. Finally, we generate the final image by the CRN, in which we propose a TAM to introduce knowledge information into the visual representations of images. The KG consists of a large amount of commonly-seen objects as entities and relationships between them, which can be formulated as triplets $T = \{(l_i, r_p, l_j)\} \in \mathcal{C} \times \mathcal{R} \times \mathcal{C}$, where $\mathcal{C} = \{l_1, l_2, \cdots, l_N\}$ is the set of object categories, and $\mathcal{R} = \{r_1, r_2, \cdots, r_P\}$ is the set of relationships. A triplet $t = (l_i, r_p, l_j) \in T$ represents that the head entity l_i and the tail entity l_j have a relationship of r_p. The input object labels are denoted as $O = \{o_1, o_2, \cdots, o_n | o_i \in \mathcal{C}\}$.

3.2 Scene Graph Selection Module

As shown in Figure 2(a), given object labels O, the SGSM searches in the KG and randomly selects a series of triplets $\{t_{ij}^p = (o_i, r_p, o_j)\}$ in which both head entities o_i and tail entities o_j belong to O. These triplets form a directed graph, also known as the scene graph $G \in (O, E)$, where nodes O represents object labels and $E \subseteq O \times R \times O$ is a set of directed edges of the form (o_i, r_p, o_j). Note that the number of selected triplets can be specified by users. Because

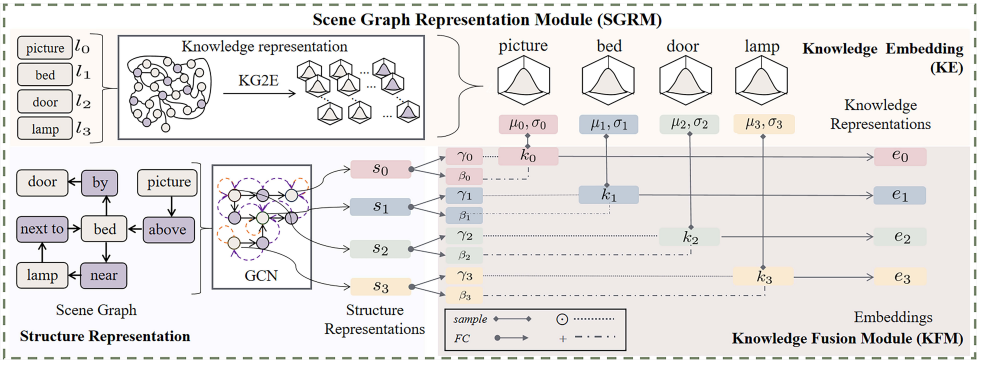


Figure 3: SGRM. The scene graph is fed into a GCN to output structure representations of objects. Based on the KG and labels, knowledge embeddings of objects and relationships are learned by KG2E. KFM integrates the two features.

of the randomness, we can obtain various scene graphs. Figure 2(b) demonstrates such scene graphs generated by the SGSM based on the same set of object labels.

3.3 Scene Graph Representation Module

The goal of SGRM is to extract features of the scene graph. As shown in Figure 3, we learn the structure information and knowledge embeddings of the scene graph, and combine the two representations by a KFM. We design the following 3 branches.

Structure Representation. A series of graph convolutional layers [14] are employed to encode the scene graph $G = (O, E)$. Given the initial vectors $v_i, v_j, v_p \in \mathbb{R}^{in}$ for objects $o_i, o_j \in O$ and relationship $(o_i, v_p, o_j) \in E$, the output features $v_i', v_j', v_p' \in \mathbb{R}^D$ are computed by considering the neighborhood features along edges. By pooling candidate features of head entities and tail entities, we obtain the vector $\mathbf{s}_i \in \mathbb{R}^D$ to represent the structure information of the object o_i.

Knowledge Embedding (KE). We employ KG2E [8] to learn knowledge embeddings of objects and relationships from the KG. Given the KG $T = \{(l_i, r_p, l_j)\}$, KG2E represents entities and relationships as high-dimensional Gaussian distributions, namely $\mathcal{P}_i^l \sim \mathcal{N}(\mu_i^l, \Sigma_i^l)$ for entity l_i, and $\mathcal{P}_p^r \sim \mathcal{N}(\mu_p^r, \Sigma_p^r)$ for relationship r_p, where $\mu \in \mathbb{R}^D$ is the mean vector and $\Sigma \in \mathbb{R}^{D \times D}$ is the covariance matrix (actually replaced with the diagonal covariance). To introduce extra information beyond objects in the scene graph, we randomly sample from the knowledge representation $\mathcal{N}(\mu_i^l, \Sigma_i^l)$ and get $\mathbf{k}_i \in \mathbb{R}^D$ to represent the knowledge embedding of object o_i. For the relationship r_p, we use the mean vector μ_p^r as its feature embedding.

Knowledge Fusion Module (KFM). The KFM fuses the structure representation and knowledge embedding of each object. Since structure representations are derived from the scene graph, and knowledge embeddings which are learned in the KG incorporate extra information beyond the objects, the combination of these two features can lead to more powerful representations. In this module, we transform the knowledge embedding by the guidance of the structure representation $\mathbf{s}_i$. Concretely, the structure representation is first encoded by two fully-connected layers, respectively, and then integrated by a linear system,

$$\gamma_i = f_\gamma(\mathbf{s}_i), \beta_i = f_\beta(\mathbf{s}_i), \tag{1}$$

$$\mathbf{e}_i = \gamma_i \odot \mathbf{k}_i + \beta_i, \tag{2}$$

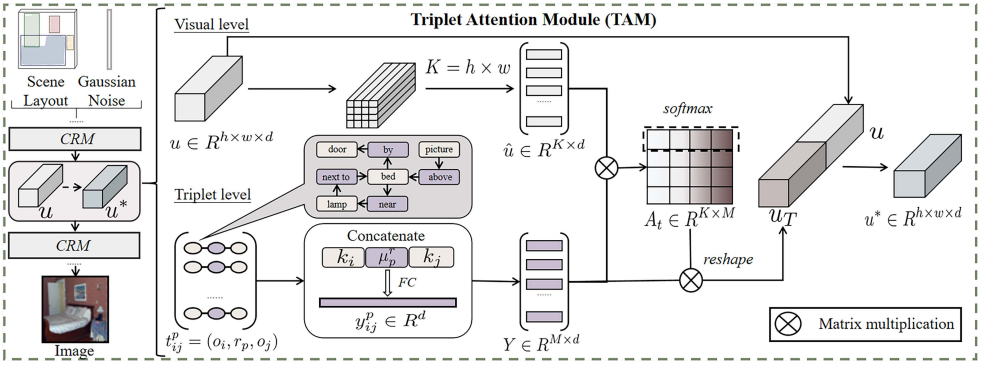


Figure 4: Triplet Attention Module. Visual level: the output feature is segmented from the spatial dimension into several image sub-region features. Triplet level: the knowledge information of the three elements in the triplet is fused to obtain the triplet features. Finally, the visual feature is updated by knowledge information of triplet in conjunction with attention mechanisms.

where $\odot$ means element-wise product and $\gamma_i, \beta_i \in \mathbb{R}^D$.

3.4 Layout Prediction

Based on the integrated features output by the KFM, a scene layout is predicted by learning the bounding box and mask of each object. Following [14], we feed the integrated representation $\mathbf{e}_i$ of the object o_i in a box prediction network and a mask prediction network to give a bounding box $b_i = (x_b, y_b, x_t, y_t)$ and a soft binary mask m_i of shape $m \times m$. The object layout is obtained via warping the mask embedding to the position of the bounding box by the bilinear interpolation operation. We sum all the object layouts to give a scene layout.

3.5 Image Generation

Given the scene layout and Gaussian noise, an image is generated and refined by a CRN [3] in a coarse-to-fine manner. The CRN is composed of a set of Cascaded Refinement Modules (CRMs). Each CRM takes as input the channel-wise concatenation of the scene layout and the feature maps of the previous module. Then the input is fed into a pair of convolutional layers. The output feature map is passed to an up-sampling layer and input to the next CRM.

Triplet Attention Module (TAM). To improve the semantic consistency between the label domain and the image domain, we propose a TAM to integrate the triplet knowledge embeddings and the visual features produced by the CRM, shown in Figure 4. On the visual level, we denote the output visual feature from the CRM as $\mathbf{u}$ with the size $h \times w \times d$. We then reshape it according to the column dimension and get the visual matrix $\hat{\mathbf{u}}$ with the size $K \times d$, where $K = h \times w$. On the triplet level, we first concatenate the knowledge embeddings $\mathbf{k}_i, \mu^r_p, \mathbf{k}_j$ of the triplet $\{t^p_{ij} = (o_i, r_p, o_j)\}$ and then input it in a fully-connected layer to output a knowledge vector $\mathbf{y}^p_{ij}$,

$$\mathbf{y}^p_{ij} = FC([\mathbf{k}_i, \mu^r_p, \mathbf{k}_j]). \tag{3}$$

We aggregate the knowledge vectors of all triplets to obtain the knowledge matrix Y with the size $M \times d$, where M is the number of triplets in the scene graph. Then a visual-knowledge affinity matrix $A \in \mathbb{R}^{K\times M}$ is computed by a matrix multiplication operation

$$\mathbf{A} = \hat{\mathbf{u}} \times \mathbf{Y}^T. \tag{4}$$

We process it with a $softmax$ function to get the attention matrix A_t in which the summation of each row equals to 1. Therefore, the element $a^t_{uv} \in A_t$ represents the attention that the u-th visual feature pays to the v-th triplet. Subsequently, the visual representation is transformed under the guidance of the knowledge embedding of triplets

$$\mathbf{u}_T = \mathbf{A}_t \times \mathbf{Y}, \tag{5}$$

and reshaped to the size of $h \times w \times d$. Then the concatenation feature $[\mathbf{u}_T, \mathbf{u}]$ is processed by a 1×1 convolutional layer to get $\mathbf{u}^* \in R^{h \times w \times d}$, which is finally input to the next CRM.

3.6 Training

We train two discriminators D_{img} and D_{obj} adversarially against the generation network, similar to sg2im [14]. SGSM does not participate in training. Given the scene graphs, the rest components of Label2im including SGRM, layout prediction, and image generation network are trained in an end-to-end manner. We use ground truth scene graphs during training and use SGSM to generate scene graphs at test time. The generation network minimizes the following objective function which consists of 7 losses,

$$\mathcal{L} = \lambda_1 \mathcal{L}_{pix} + \lambda_2 \mathcal{L}_{box} + \lambda_3 \mathcal{L}^{img}_{GAN} + \lambda_4 \mathcal{L}^{obj}_{GAN} + \lambda_5 \mathcal{L}^{obj}_{AC} + \lambda_6 \mathcal{L}^{img}_{p} + \lambda_7 \mathcal{L}^{obj}_{p}, \tag{6}$$

where *pixel loss* $\mathcal{L}_{pix} = \|I - \hat{I}\|_1$ is the L_1 loss between the ground truth image and the generated image; *box loss* $\mathcal{L}_{box} = \sum_{i=1}^{n} \|b_i - \hat{b}_i\|$ is the L_1 loss between the ground truth boxes and predicted boxes; *image adversarial loss* L^{img}_{GAN} and *object adversarial loss* L^{obj}_{GAN} are from the discriminators D_{img} and D_{obj}, respectively; *auxiliary classifier loss* $\mathcal{L}^{obj}_{AC}$ ensures each generated object to be classified by D_{obj}; *image perceptual loss* $\mathcal{L}^{img}_{p}$ is the cosine distance between the features of ground truth image and the generated image, similar to [19]; *object perceptual loss* $\mathcal{L}^{obj}_{p}$ is the cosine distance between the features of the ground truth object crop and the generated object, similar to $\mathcal{L}^{img}_{p}$; and coefficient $\lambda_1, \lambda_2, \cdots, \lambda_7$ are hyper-parameters. Note that the first 5 losses are consistent with sg2im.

4 Experiments

4.1 Experiment Settings

Dataset. We train and test the proposed method on the Visual Genome [17]. Visual Genome incorporates a large amount of triplet annotations of different object categories from various realistic scenes. Based on these triplet annotations, we construct a large-scale KG and adopt the preprocessing and data split strategy of [14].

Implementation Details. The feature dimension is set to $D = 128$ in Section 3.3. The hyper-parameters $\lambda_1, \lambda_2, \cdots, \lambda_7$ in Equation 6 are 1, 10, 1, 1, 0.1, 0.5 and 1. Our model is optimized by the Adam algorithm [15] with learning rate 10^{-4} and batch size 32 for 1 million iterations. All experiments are conducted on a single GeForce RTX 2080 Ti GPU. According to the settings of previous works [14, 19, 42], we train our model to generate 64×64 images.

Evaluation Metrics. We employ four commonly-used metrics to evaluate the performance of compared methods, including Inception Score (IS) [28], Fréchet Inception Distance (FID) [9], Diversity Score (DS) [41], and Classification Accuracy (AC) [43]. More implementation details and descriptions of metrics can be found in the supplementary material.

Figure 5: Visual results of sg2im [14] and Label2im on the Visual Genome.

Method	IS↑	FID↓	DS↑	AC↑
Real Imgs	13.9±0.5	-	-	50.16
sg2im[14]	5.5±0.1	71.27	0.12±0.06	32.72
PasteGAN[19]	6.9±0.2	58.53	0.24±0.09	-
Label2im-sg	**7.4±0.2**	**43.49**	0.11±0.13	**39.75**
Label2im	7.2±0.2	44.61	**0.35±0.07**	33.65
sg2im(GT)	6.3±0.2	52.96	0.15±0.12	43.44
Layout2im[42]	8.1±0.1	**31.25**	0.17±0.09	48.09
PasteGAN(GT)	8.2±0.2	35.25	**0.29±0.08**	-
Label2im(GT)	**8.4±0.2**	37.97	0.08±0.10	**48.99**

Table 1: Performance of the evaluated methods on the Visual Genome. Label2im-sg means providing ground truth scene graphs during test time. GT means using ground truth bounding boxes.

4.2 Comparison with State-of-the-Arts

We compare the proposed method with 3 state-of-the-art scene generation models, including sg2im [14], PasteGAN [19], and Layout2im [42].

Quantitative Evaluation. As shown in Table 1, for fairness, if we provide ground truth scene graphs at test time, our method (Label2im-sg) performs favorably against the compared methods on most metrics except for the DS, which indicates that our method is inferior in generating diverse images. This may be because that we employ the KG2E to extract knowledge representations of objects. This method models the certainty of entities and relationships, and thus degrades the diversity of our generation model. However, we can solve this problem by randomly generating different scene graphs by the SGSM (see Label2im in Table 1). Note that both PasteGAN and Layout2im extract appearance information from real image crops, while we compensate information from the KG with non-image modalities. Our method achieves competitive or even better performance with these models given ground truth bounding boxes (see Label2im(GT) in Table 1).

Qualitative Evaluation. The visual results in Figure 5 show that our method improves image quality and is more sensitive to relationships in the scene graph. Such as Figure 5(e), images generated by our method is more fitting to the triplet $(man, above, snow)$. This attributes to the knowledge representation learning in the KG which makes our method better understand the interactions between objects. The improvement in image quality from given ground truth layout also demonstrates the effectiveness of the TAM. The two scene graphs (Generated Graph(A and B) in Figure 5) output by the SGSM and the generated images (Label2im(A and B)) indicate that our method can generate diverse images. More visual results are shown in the supplementary material.

4.3 Ablation Study

Given ground truth scene graphs, we conduct ablation studies to verify the necessity of SGRM and TAM. As shown in Table 2, compared to sg2im, **Baseline** further adopts the image and object perceptual loss $\mathcal{L}_p^{img}$ and $\mathcal{L}_p^{obj}$ (Equation 6). In **+KE+Cat**, based on the **Baseline**, we further learn knowledge embeddings of scene graphs, and replace the **KFM** with the concatenation operation. In **+KE+KFM+Global**, to replace **TAM**, inspired by [31], we extract a global vector from the scene graph and concatenate it with the layout in the image generation process. **TAM w/o Rel** omits the relationship features in the **TAM**. **w/o** $\mathcal{L}_p^{img}$

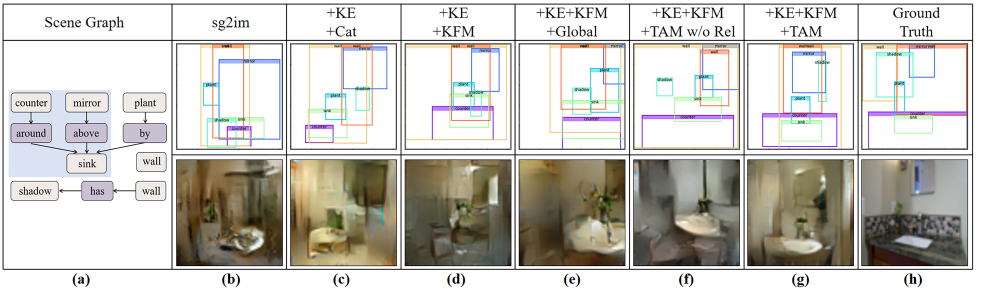


Figure 6: Generated images given scene graphs in the ablation study.

Method	IS↑	FID↓
Real Imgs	13.9±0.5	-
Sg2Im[14]	5.5±0.1	71.27
Baseline	6.1±0.2	53.86
+KE+Cat	6.4±0.1	51.64
+KE+KFM	6.7±0.1	50.93
+KE+KFM+Global[31]	6.8±0.2	50.75
+KE+KFM+TAM w/o Rel	7.1±0.2	44.34
+KE+KFM+TAM w/o $\mathcal{L}_p^{img}$	6.5±0.1	53.55
+KE+KFM+TAM w/o $\mathcal{L}_p^{obj}$	7.3±0.2	44.06
+KE+KFM+TAM (Ours)	**7.4±0.2**	**43.49**

Table 2: Ablation Study on IS and FID. ↑ means higher is better; ↓ means lower is better.

and **w/o** $\mathcal{L}_p^{obj}$ omit the image and object perceptual loss $\mathcal{L}_p^{img}$ and $\mathcal{L}_p^{obj}$ respectively during training phase.

+**KE**+**Cat** outperforms the **Baseline**, which verifies that knowledge information contributes to the representations of scene graphs. Compared with +**KE**+**Cat**, the better performance of +**KE**+**KFM** indicates that it is beneficial to integrate structure information and knowledge embeddings adaptively. Comparing +**KE**+**KFM** +**Global** and +**KE**+**KFM**+ **TAM**, it is more effective to employ the TAM to guide the generation process. Comparing +**KE**+**KFM**+ **TAM w/o Rel** and +**KE**+**KFM**+**TAM**, it shows that relationship information is important to image generation. The visual results in Figure 6 prove that knowledge embedding is necessary for image details and TAM enhances semantic consistency between images and triplets (note the part marked in blue in the scene graph).

5 Conclusion

In this paper, we propose an image generation method from object labels, called Label2im. To ease and diverse the access to relationship interactions between labels, we generate possible scene graphs by the SGSM. To fill the gap between the semantic domain and the image domain, we propose to learn and integrate structure information and knowledge embeddings of the scene graph in SGRM. In the image generation process, we propose a TAM to introduce knowledge representation to ensure the semantic consistency. Extensive experiments on the Visual Genome demonstrate that our method is able to generate realistic images given object labels and better respects the scene graph.

Acknowledgments

Supported by the Ministry of Science and Technology of the People's Republic of China (No. 2018AAA0102003), the Fundamental Research Funds for the Central Universities (No. 82232006), and the National Natural Science Foundation of China (No. 61902053, No. 61976040, No. 62172073).

References

[1] Oron Ashual and Lior Wolf. Specifying object attributes and relations in interactive scene generation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4561–4569, 2019.

[2] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in Neural Information Processing Systems*, 26, 2013.

[3] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1511–1520, 2017.

[4] Dieter Fensel, Umutcan Şimşek, Kevin Angele, Elwin Huaman, Elias Kärle, Oleksandra Panasiuk, Ioan Toma, Jürgen Umbrich, and Alexander Wahler. Introduction: What is a knowledge graph? In *Knowledge Graphs*, pages 1–10. 2020.

[5] Jon Gauthier. Conditional generative adversarial nets for convolutional face generation. *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition*, 2014(5):2, 2014.

[6] Mingming Gong, Yanwu Xu, Chunyuan Li, Kun Zhang, and Kayhan Batmanghelich. Twin auxiliary classifiers gan. *Advances in Neural Information Processing Systems*, 32:1328, 2019.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2014.

[8] Shizhu He, Kang Liu, Guoliang Ji, and Jun Zhao. Learning to represent knowledge graphs with gaussian embedding. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 623–632, 2015.

[9] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems*, 30, 2017.

[10] Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7986–7994, 2018.

[11] Maor Ivgi, Yaniv Benny, Avichai Ben-David, Jonathan Berant, and Lior Wolf. Scene graph to image generation with contextualized object layout refinement. In *IEEE International Conference on Image Processing*, pages 2428–2432, 2021.

[12] Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 687–696, 2015.

[13] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3668–3678, 2015.

[14] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1219–1228, 2018.

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015.

[16] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[17] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[18] Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. Object-driven text-to-image synthesis via adversarial training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12174–12182, 2019.

[19] Yikang Li, Tao Ma, Yeqi Bai, Nan Duan, Sining Wei, and Xiaogang Wang. Pastegan: A semi-parametric method to generate image from scene graph. *Advances in Neural Information Processing Systems*, 32:3948–3958, 2019.

[20] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[21] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729, 2008.

[22] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *International Conference on Machine Learning*, pages 2642–2651, 2017.

[23] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2337–2346, 2019.

[24] Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508, 2017.

[25] Mengshi Qi, Yunhong Wang, Jie Qin, and Annan Li. Ke-gan: Knowledge embedded generative adversarial networks for semi-supervised scene parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5237–5246, 2019.

[26] Xiaojuan Qi, Qifeng Chen, Jiaya Jia, and Vladlen Koltun. Semi-parametric image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8808–8816, 2018.

[27] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, pages 1060–1069, 2016.

[28] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in Neural Information Processing Systems*, 29:2234–2242, 2016.

[29] Wei Sun and Tianfu Wu. Image synthesis from reconfigurable layout and style. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10531–10540, 2019.

[30] Hongchen Tan, Xiuping Liu, Meng Liu, Baocai Yin, and Xin Li. Kt-gan: knowledge-transfer generative adversarial network for text-to-image synthesis. *IEEE Transactions on Image Processing*, 30:1275–1290, 2020.

[31] Subarna Tripathi, Anahita Bhiwandiwalla, Alexei Bastidas, and Hanlin Tang. Heuristics for image generation from scene graphs. *ICLR workshop*, 2019.

[32] Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International Conference on Machine Learning*, pages 1747–1756, 2016.

[33] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[34] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

[35] Han Xiao, Minlie Huang, Yu Hao, and Xiaoyan Zhu. Transa: An adaptive approach for knowledge graph embedding. *arXiv preprint arXiv:1509.05490*, 2015.

[36] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1316–1324, 2018.

[37] Guojun Yin, Bin Liu, Lu Sheng, Nenghai Yu, Xiaogang Wang, and Jing Shao. Semantics disentangling for text-to-image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2327–2336, 2019.

[38] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Bridging knowledge graphs to generate scene graphs. In *European Conference on Computer Vision*, pages 606–623, 2020.

[39] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5907–5915, 2017.

[40] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1947–1962, 2018.

[41] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018.

[42] Bo Zhao, Lili Meng, Weidong Yin, and Leonid Sigal. Image generation from layout. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8584–8593, 2019.

[43] Han Zhao, Shanghang Zhang, Guanhang Wu, Geoffrey J Gordon, et al. Multiple source domain adaptation with adversarial learning. *ICLR workshop*, 2018.