Supplementary material for 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

1 Implementation Details

1.1 Data Processing

We conduct our experiments on the Visual Genome (VG) dataset [7] which contains a total of 108,077 images, each annotated with a scene graph. We process the data in the VG following sg2im [6]. Specifically, the data are divided into 80% training set, 10% validation set, and 10% test set. We calculate the occurring frequency of objects and relationships in the training set, and reserve the top 178 objects and 45 relationship types. Small objects are discarded, and each image incorporates $3 \sim 30$ objects and at least one relationships. The scene graph of each image is updated according to the above process. We then collect all textual triplets (*head entity*, *relationship*, *tail entity*) of scene graphs to form a Knowledge Graph (KG) which stores the interactions between objects in common scenarios. As shown in Figure 1, we also use the graph database, Neo4j, to store and visualize the KG. Table 1 shows some attributes of VG.

Item	♯Train	♯Val	♯Test	♯Objs	♯Relations	♯Triplets
Num	62,565	5,506	5,088	178	45	333,047

Table 1: Attributes of VG. ♯Objs denotes the number of object categories. ♯Relations denotes the number of relationship categories. ♯Triplets denotes the number of all triplets after data pre-processing.

* indicates the corresponding authour.

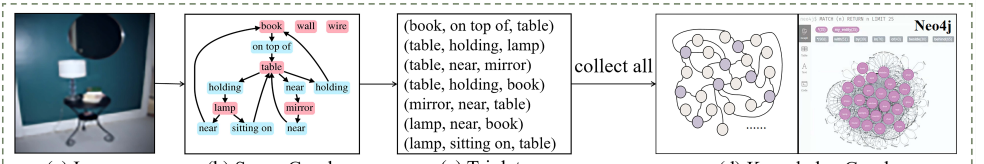

Figure 1: The process of building a KG. The left figure in (d) is the abstraction used in our paper, while the right one is the visual representation of the KG using Neo4j.

1.2 Evaluation Metrics

We employ four metrics to evaluate the performance of the proposed method and the state-of-the-arts.

Inception Score (IS). IS [10] is widely used to measure the quality and diversity of generated images. Like previous works [8, 11], we employ the pre-trained network, Inception-V3 [13], to extract image features and compute the IS. A higher IS is better.

Fréchet Inception Distance (FID). FID [5] measures the distance between the distributions of generated images and ground truth. A lower FID means the generated images are more similar to the real images.

Diversity Score (DS). DS [16] reflects the diversity of generated images by evaluating the perceptual similarity between a pair of images given the same input. A higher DS is better.

Classification Accuracy (AC). AC measures the quality of each object in the generated images through a classification network. As described in [1, 17], we first crop object patches from ground truth images and then employ the Resnet-101 [3] to train an object classification network based on them. During testing, we compute the classification accuracy for objects in the generated images. Higher AC means the generated objects are more recognizable.

1.3 More Details of Ablation Study

In Section 4.3 of the manuscript, to further validate the effectiveness of our method, we design several baseline networks for ablation studies. In this subsection, we describe +**Cat**, +**Global**, and +**TAM w/o Rel** in details.

In particular,+**Cat** concatenates the knowledge vector $k_i \in \mathbb{R}^D$ and the structure representation $s_i \in \mathbb{R}^D$ to obtain a vector of dimension $2D$, which is then mapped to a D-dimensional vector e_i; +**Global** introduces global knowledge information of the scene graph into the generation process, inspired by [14]. Given a scene graph, all knowledge vectors are pooled and transformed by a fully-connected layer to output a global vector with dimension $\frac{D}{4}$. Then, the global vector is expanded to the size of $H \times W \times \frac{D}{4}$, in which $H \times W$ is the resolution of the generated image. Finally, the global tensor is concatenated with the scene layout and fed into the CRN to generate the image; +**TAM w/o Rel** integrates visual features output by the CRM and knowledge embeddings of objects, ignoring relationships in triplets. Without considering the information of relationships, we directly input knowledge embeddings of objects to a fully-connected layer and aggregate them to obtain the knowledge matrix Y with size $n \times d$, where n is the number of objects in the scene graph, d is the channel dimension of the feature output by CRM. Other operations remain the same as TAM.

Method	IS↑	FID↓
Real Imgs	20.5±1.5	-
sg2im*[6]	8.1±0.2	59.94
Label2im*	12.5±0.2	36.72
sg2im+LostGAN-v2 [12]	9.0±0.1	35.70
Label2im+LostGAN-v2 [12]	9.5±0.3	34.71
sg2im*(GT)	8.3±0.2	56.17
LostGAN-v2 [12]	10.7±0.3	29.00
Label2im*(GT)	13.7±0.2	34.32

Table 2: Performance of the 128 × 128 images generated by the evaluated methods on VG. In particular, sg2im* means the reproduced model for 128 × 128 images. Label2im* means our model to generate images at the resolution of 128 × 128.

2 Additional Experiments and Discussion

2.1 Discussion of High Resolution

Similar to [6], we employ a Cascaded Refinement Network (CRN) [2] consisting 5 Cascaded Refinement Modules (CRMs) to generate 64 × 64 images (reported in the manuscript). In this subsection, we conduct preliminary experiments on the fine-tuned version of Label2im to generate 128 × 128 images, reported in Table 2. Based on the supplementary material of sg2im [6], we reproduce the high-resolution version of it to generate 128 × 128 images, named sg2im* in Table 2. Specifically, we add an extra CRM to CRN and a convolutional layer to each discriminator. For our method, we apply the SGRM and TAM (applied after 2nd, 3rd, 4th, 5th CRM) into sg2im*, named as Label2im* in Table 2.

In addition, Table 2 also presents the results of replacing the layout-to-image generator with LostGAN-v2 [12]. Specifically, sg2im+LostGAN-v2 means using sg2im to predict layouts and LostGAN-v2 to generate images. Label2im+LostGAN-v2 means using Label2im to predict layouts from scene graphs and LostGAN-v2 to produce images. Compared to the baseline method, Label2im* achieves great improvement in three cases: giving scene graphs, replacing the generator, and giving ground truth boxes(GT). Label2im* achieves impressive performance to generate images at the resolution of 128 × 128 on IS. Figure 2 shows some visual results of Label2im*.

2.2 Discussion of Triplet Attention Module

Taking the resolution of 64 as an example, we discuss where to apply Triplet Attention Module (TAM) in this subsection. As shown in Figure 3, TAM can be flexibly applied after each CRM layer, with a total of 5 positions available (layer 0, 1, 2, 3, 4). In order to explore a reasonable application plan for TAM, we set up several experiments and the results are listed in Table 3.

In Table 3, No.1 denotes the baseline method without TAM. From No.2 to No.4, the performance of the generation models improves with the increase of TAMs, and the network with three TAMs applying after the 4th, 3rd, and 2nd CRM achieves the best performance. However, we find that the performance does not get better if we further introduce TAMs to shallower layers, *e.g.* **layer1** and **layer0**. It may be because that visual features in shallower layers lack semantic information and thus have large gaps between knowledge representa-

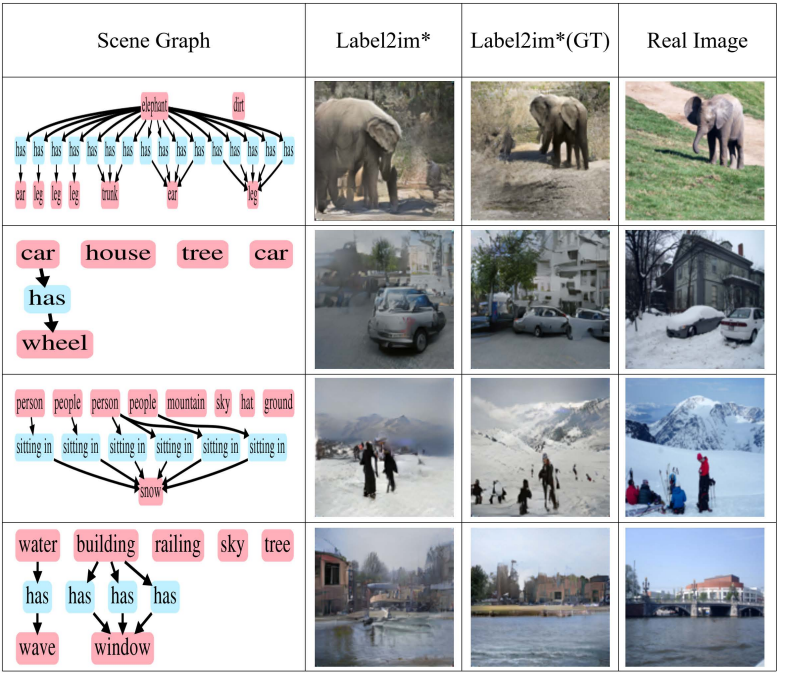

Figure 2: Visual results of Label2im* to generate 128×128 images from the given scene graphs. GT means using ground truth boxes.

tions. Therefore, in our experiments, we apply TAMs in the last three CRMs to generate images at the resolution of 64.

2.3 Discussion of KG

The KG for scene generation needs to store the interactions between objects in the form of triplets. In this subsection, we also filter and extract the triplets from the VRD [9] dataset (widely used for visual relation detection [15]). Depending on these different sources of KG, we train KG2E [9] separately to get knowledge embeddings of the objects and the relationships.

Since the requirement of the knowledge corresponding to the labels and relationships in SGRM and TAM, it is difficult to test on VG directly when the source of knowledge is

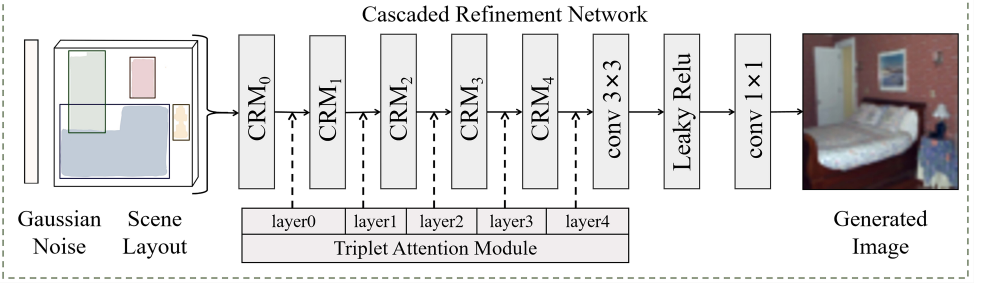

Figure 3: The optional positions in CRN for TAMs.

No.	layer4	layer3	layer2	layer1	layer0	IS↑	FID↓
1	-	-	-	-	-	6.7±0.1	50.93
2	✓	-	-	-	-	6.9±0.1	48.64
3	✓	✓	-	-	-	7.2±0.2	44.89
4	✓	✓	✓	-	-	**7.4±0.2**	**43.49**
5	✓	✓	✓	✓	-	7.3±0.1	44.21
6	✓	✓	✓	✓	✓	7.3±0.2	43.62

Table 3: Performance evaluation of TAM applied in different positions in terms of IS and FID. ✓ indicates that TAM is used after the CRM in the CRN.

Dataset	♯Objs	♯Relations	♯Triplets
VG	178	45	333047
VRD	100	70	30355
Source of knowledge	**Source of images for train**	**Source of scene graphs for test**	**IS ↑**
VG	VG	Intersection-200	3.6±0.2
VRD	VG	Intersection-200	3.2±0.4
VG+VRD	VG	Intersection-200	3.5±0.2
VG+VRD	VG	VG	7.2±0.2
VG (Ours)	VG	VG	7.4±0.2

Table 4: The upper part of the table shows the statistics of the datasets. ♯Objs denotes the number of object categories. ♯Relations denotes the number of relationship categories. ♯Triplets denotes the number of triplets we collect. The bottom of the table shows the results of 64×64 images generated from the scene graphs. Source of knowledge denotes the source of the triplets to obtain the knowledge embeddings using KG2E. Source of images for train means on which dataset the Label2im model is trained. As for the source of scene graphs for testing, Intersection-200 contains 200 scene graphs, randomly constructed.

VRD. Therefore, we construct another test set based on the intersection of the VG and VRD datasets. The intersection contains 65 objects and 16 relationships. Each time, we randomly select 5 objects from 65 objects as the label set, and then used SGSM to randomly generate 20 scene graphs from the selected labels. After 10 operations in this way, 200 scene graphs are obtained to be the test set for this experiment, named Intersection-200. We mainly focus on the 64×64 images. Table 4 shows the quantitative results.

As shown in Table 4, IS slightly decreases when the source of knowledge changes from VG to VG+VRD. This may be due to the low number of triplets from VRD. Specifically, during knowledge representation learning, VRD increases the number of object categories but fails to provide enough triplets to learn the knowledge information of the increased part. This leads to insufficient overall knowledge learning. In future work, we can construct a more adequate and large KG to learn more effective knowledge representations.

2.4 Discussion of SGSM

We imitate the user's usage process to generate images from the given labels (light, bed, door, window). As shown in Figure 4 (a) to (f), the scene graphs generated by SGSM are

mostly reasonable. Figure 4 (g) shows the case of the unreasonable scene graph. The reason for the unreasonableness is the contradiction between the two relationships (with and above) in terms of the spatial layout. In future work, we can focus more on the spatial location of different relationships and learn some rules to constrain SGSM for better scene graph selection.

3 More Visual Results

In this section, we show more visual results at the resolution of 64.

User Interface of Label2im. As shown in Figure 5, Label2im provides users with convenient and diverse options for generating images from labels. First, users pick out desired objects and specify the number of relationships and generated images. Then, the system automatically constructs scene graphs respecting the label set. Finally, users can obtain a diverse of realistic images.

Qualitative Evaluation of Comparison Methods. We show more visual results of the proposed Label2im and comparison methods in Figure 6 and Figure 7. Since we can not get the whole pre-trained model of PasteGAN [8], in Figure 6 we take the same scene graphs demonstrated in the paper [8] (the first row) as input, the generated images by the proposed Label2im are illustrated in the third row. We crop the visual results of PasteGAN from the paper and show them in the second row. Note that the PasteGAN employs real object crops to provide visual features during training and test time, and our method generates images from random noises. From Figure 6, we can see that the proposed Label2im is able to generate realistic and competitive images.

Visual Results of Ablation Studies. More visual results of ablation studies (see details in Section 4.3 in the manuscript) is illustrated in Figure 8, which demonstrates the effectiveness of each main component of Label2im.

Diversity of Label2im. Figure 9 shows that the proposed Label2im is able to generate a diverse of realistic images given the same set of object labels. Given object labels, our method randomly explores possible relationships of objects from the KG and forms a series of scene graphs, which provides great potential for generating images with quite different appearances and reasonable layouts.

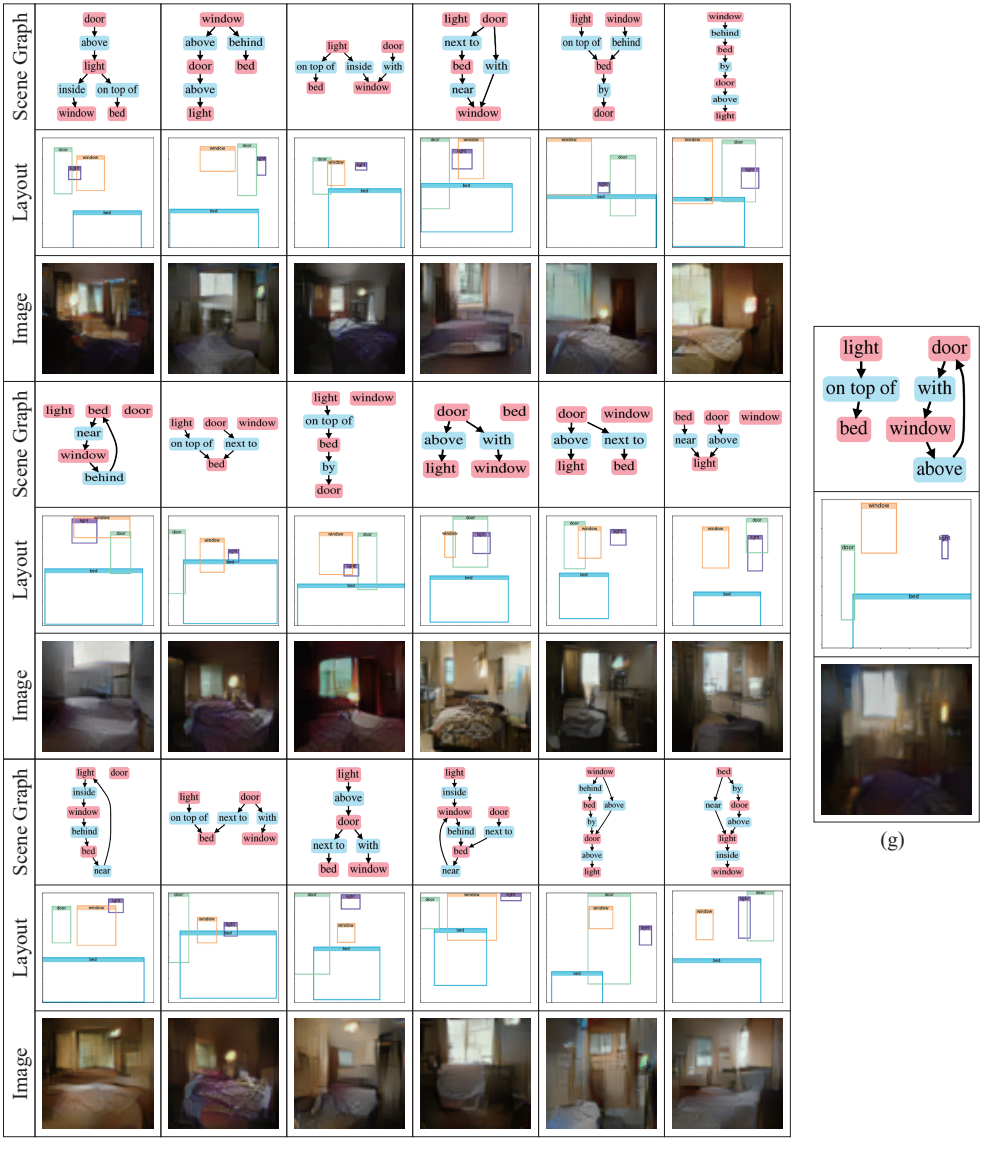

Figure 4: Random results of the given labels (light, bed, door, window).

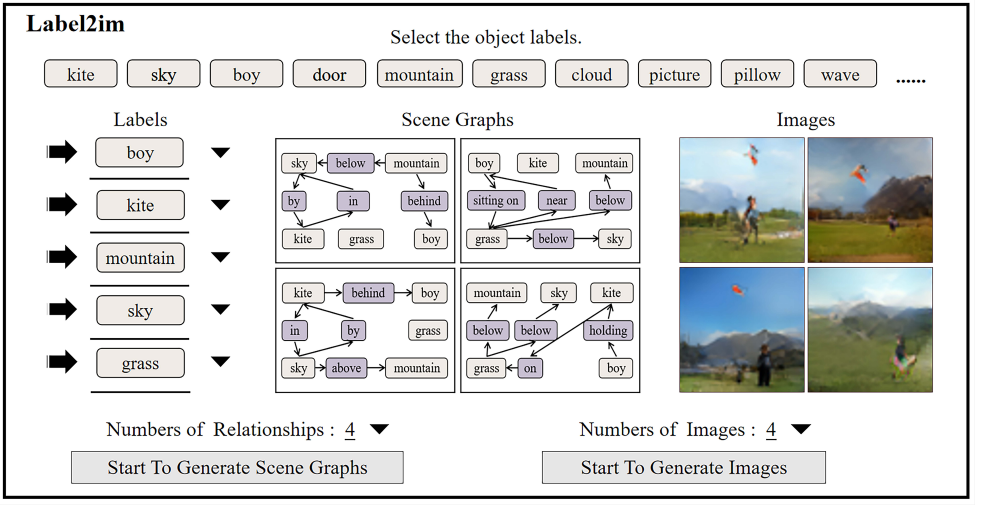

Figure 5: The interface of the proposed Label2im which allows users to specify object categories for generating images.

Figure 6: Visual evaluation of the proposed Label2im and PasteGAN [8].

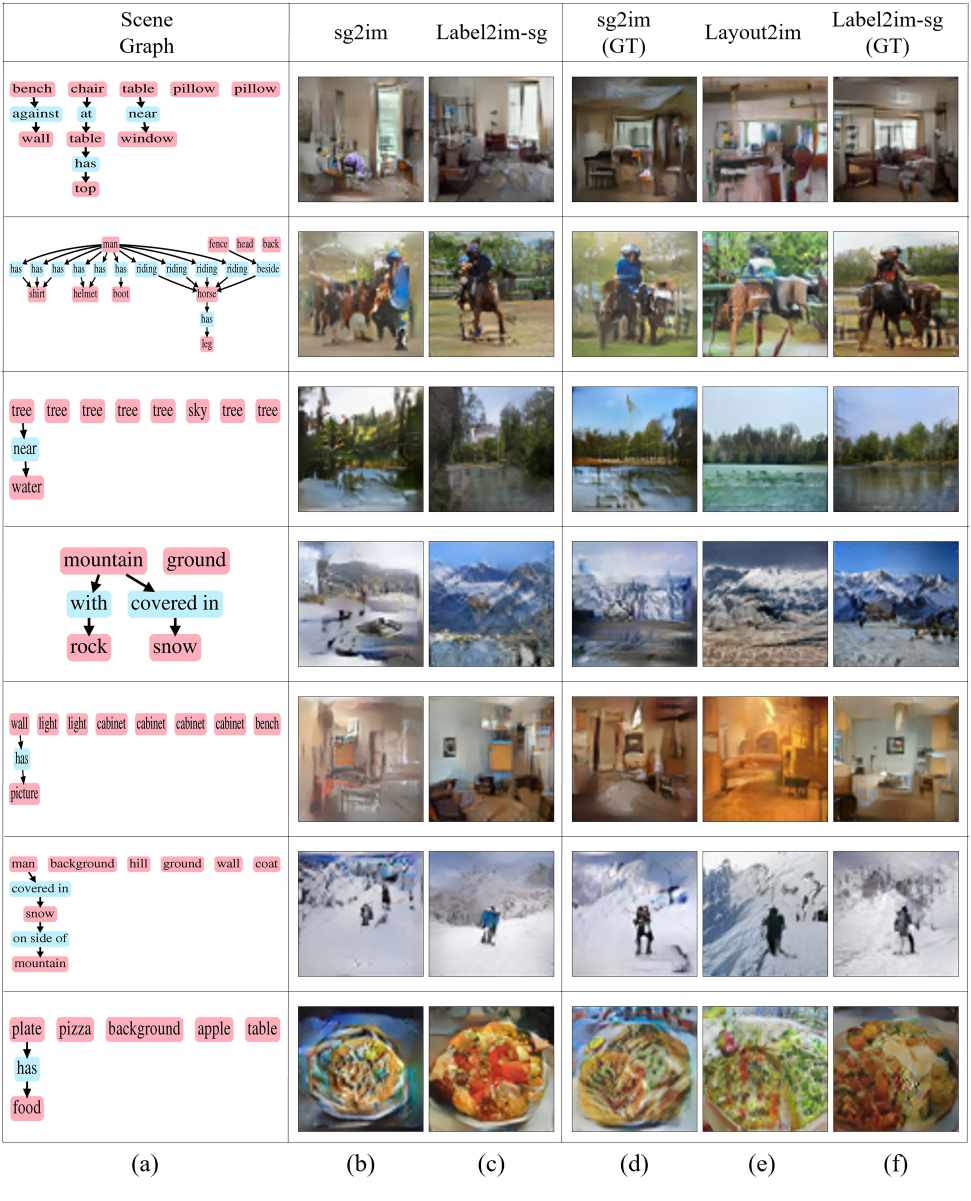

Figure 7: Qualitative evaluations of the proposed Label2im and the state-of-the-arts, including sg2im [6] and Layout2im [17]. *-sg means that using ground truth scene graphs during test time. GT means using ground truth bounding boxes of objects.

Figure 8: Visual results of ablation studies given scene graphs.

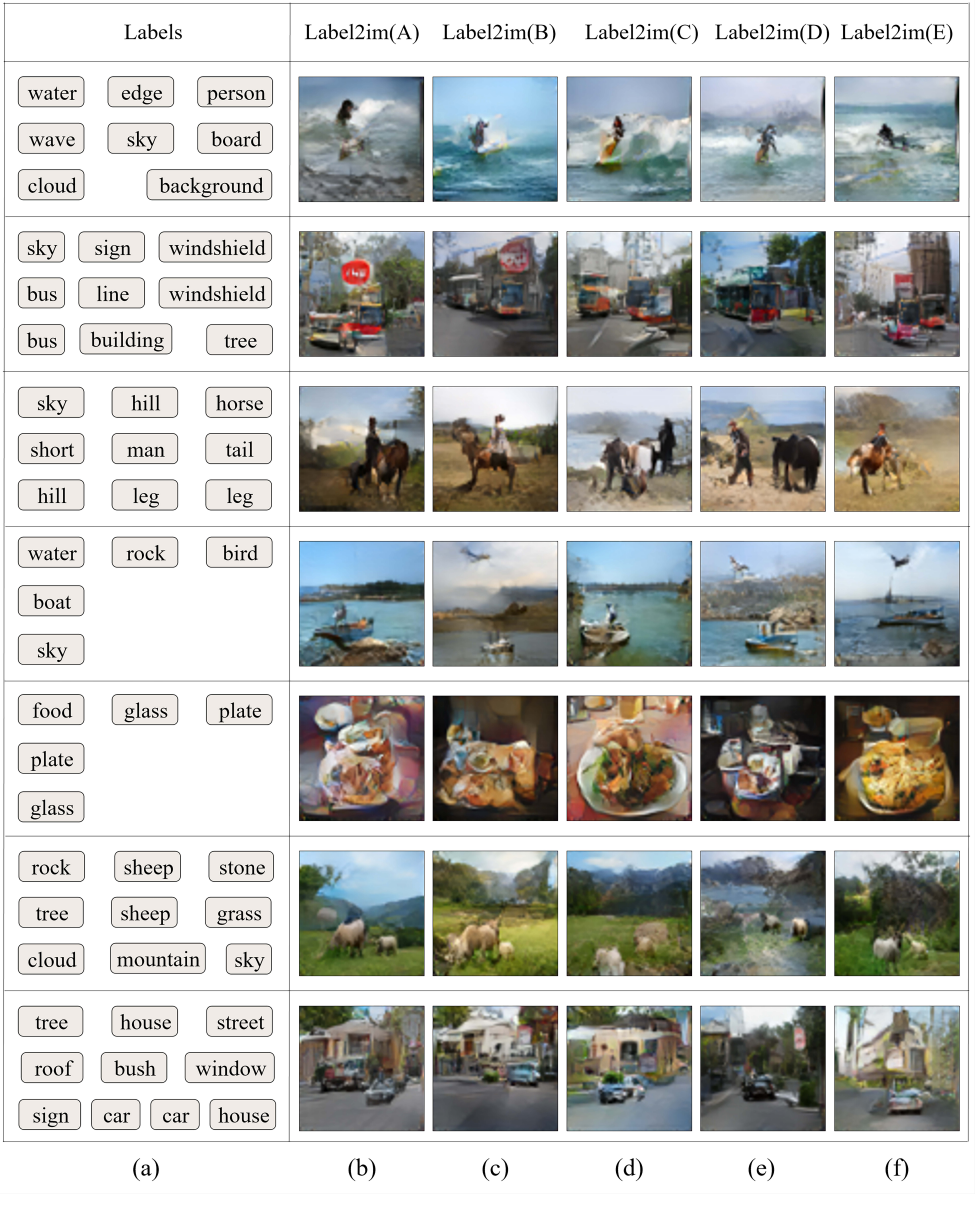

Figure 9: Diversity of Label2im. Given the same set of object labels, the proposed Label2im is able to generate a variety of realistic images.

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] 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.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[4] 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.

[5] 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. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6629–6640, 2017.

[6] 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.

[7] 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.

[8] 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.

[9] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision*, pages 852–869, 2016.

[10] 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.

[11] 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.

[12] Wei Sun and Tianfu Wu. Learning layout and style reconfigurable gans for controllable image synthesis. *arXiv preprint arXiv:2003.11571*, 2020.

[13] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

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

[15] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE Conference on Computer vision and Pattern Recognition*, pages 5532–5540, 2017.

[16] 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.

[17] 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.