M.Sc.Eng.
Masters of Science in Engineering

DTU Compute
Department of Applied Mathematics and Computer Science

Generating Images from Scene Graphs Using Diffusion Models

Yeganeh Ghamary (s194258)

Kongens Lyngby 2024

DTU Compute
Department of Applied Mathematics and Computer Science
Technical University of Denmark

Matematiktorvet
Building 303B
2800 Kongens Lyngby, Denmark
Phone +45 4525 3031
compute@compute.dtu.dk
www.compute.dtu.dk

Contents

CHAPTER 1

Introduction

1.1 Problem Formulation

Using text as conditioning in image generation models has some shortcomings. Image descriptions may be long, ambiguous, and loosely structured. This is especially important in the more complex cases with multiple objects [14]. Furthermore, some commonly used text encoders such as CLIP have some weaknesses in some complex tasks such as counting objects and capturing structural information [11]. As Johnson et al. [8] have shown, scene graphs offer a more precise representation of the image content, making them superior as a control in image generation models.

1.2 Literature Review

Recently, different models have been proposed which generate images from scene graphs. The end-to-end procedure is mostly executed in two steps: Firstly encoding the scene graph and generating the image from the embeddings. As a benchmark model, I use the Sg2Im model proposed by [8] which is tested on Visual Genome (VG) [9] and Common Objects in Context (COCO) [2] datasets. The model constructs a scene layout from predicted bounding boxes and segmentation masks of all the objects in the graph. The encoder consists of a Graph Convolusion Network which generates an embedding vector for each object, and the mask regression model and box regression network transform the embeddings into scene layout. The images are then generated using the Cascaded Refinement Network. On COCO dataset, the StackGAN outperforms the proposed model concerning Inception Score. However, the images generated by Sg2Im are semantically closer to the captions and Sg2Im generates images with more recognizable objects based on the user study. From the qualitative results, one could conclude that the structure in the scene graphs allows the sg2im model to reason about the objects and spatial relationships, which makes the model superior to text-based models in generating more complex images with recognizable objects. It should, however, be noted that their qualitative result was biased against StackGAN since the captions were not as comprehensive as the scene graphs.

Sortino et al. [14] have recently proposed a transformer-based image generator from scene graphs. Their model outperforms sg2im and PasteGAN on both COCO and VG datasets concerning Frechet Inception Distance (FID) and Inception Score (IS), and has flexibility in handling changes in the input scene graph.

Later, Farshad et al. [4] have proposed a method similar to sg2im [8] but it instead uses CLIP embeddings of nodes in the scene graph. The scene layout is then used as a guidance in a latent diffusion model. Their proposed model outperforms sg2im and PasteGAN on COCO dataset. Their qualitative results seem more accurate than the other methods, however the comparison does not seem fair due to different image resolutions.

Since the amount of scene-graph data is sparse, fine-tuning the latent diffusion models such as Stable Diffusion can be challenging. [4] On October 2023, Fundel [5] proposed multiple approaches to tackle this problem by conditioning in Latent Diffusion models, using methods such as ControlNet[18] and Gated Self-Attention. While he proposes different methods, the current results are limited to two methods on COCO dataset. First method is using ControlNet on the generated scene layouts from sg2im [8] and the second one is using the graph encodings plus the BERT text embeddings directly in Gated Salf-Attention. The results show that the sparse scene layout of sg2im is the current bottleneck for ContorlNet. While his results are limited to only 10 scene graphs and the quantitative results are not meaningful, he suggests multiple ideas for future work. Firstly he argues that the transformation

of graph encodings to scene layout which has been the method in the previous papers can cause a loss of information, especially the action, and relationships that are not spatial. Furthermore, it would be interesting to compare the performance of the current setup with ControlNet to the Stable Diffusion Model conditioned on CLIP embeddings, the proposed experiment (i) in [5]. While it is very interesting to investigate the effect of directly using graph embeddings instead of scene layouts, due to the time constraint, I will focus on only the second idea inspired by Fundel's work [5].

1.3 Purpose of the Project

The project is a proof of concept with a primary focus on investigating and evaluating the feasibility of ControlNet model in generating images conditioned on scene graphs, using a subset of Visual Genome Dataset[9], which to my knowledge is the first attempt. More concretely, this project aims to answer the following research questions:

1. How does the performance of ControlNet compare to the performance of Cascaded Refinement Network from sg2im [8] in generating the images from scene graphs using a subset of VG dataset?
2. How does the performance of ControlNet conditioned on structured scene graphs compare to the performance of Stable Diffusion conditioned on unstructured text in image generation?

There are different factors in evaluating the performance of the models, such as the quality and diversity of the generated images. It is also important that the generated image obeys the scene graph, meaning that it contains the recognizable objects and relationships of the scene graph. The evaluation is both quantitative and qualitative for the mentioned three models ControlNet, Stable Diffusion, and sg2im.

1.4 Scope of the project

The pipeline of this project consists of 3 steps:

0. Pre-processing the Dataset
1. Encoding the Scene Graph
2. Generating image

The focus of this project is on the third point, the generation of images and evaluating the performance of different generation models. Due to time constraints, I use the pre-trained models for the whole pipeline. The first two steps are the same as sg2im. This model is the only suitable option at the time of the project since the state-of-the-art mentioned in the Literature Review 1.2 have not published their pre-trained models yet. The overview of the pipeline of three models can be seen in Figure 1.1

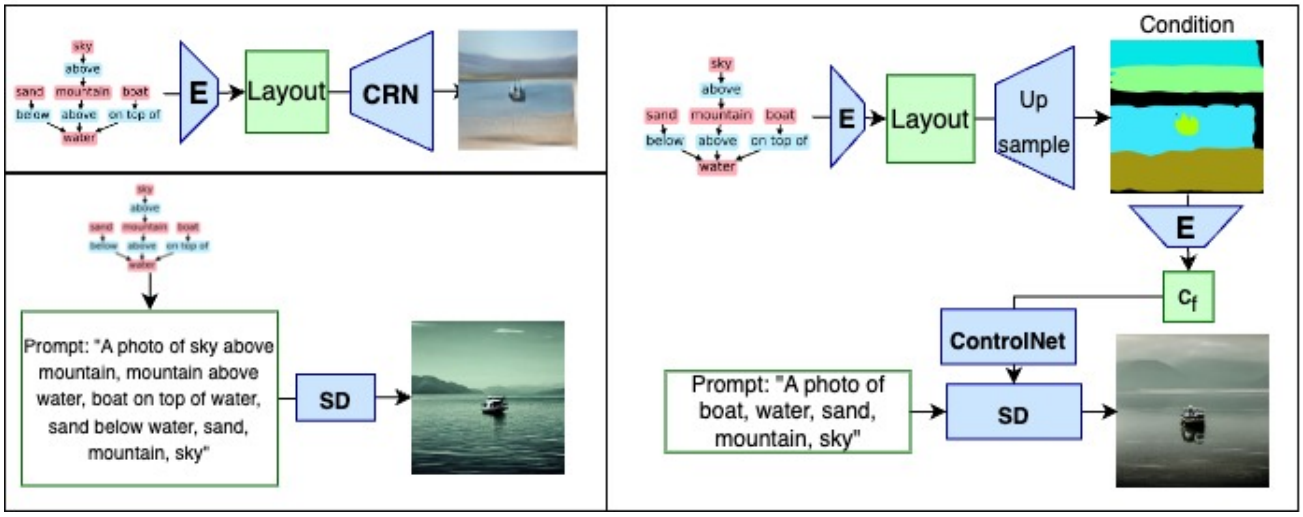


Figure 1.1: An overview of the pipelines: Top Left is the pipeline of sg2im, Below Left is the pipeline of Stable Diffusion, and Right is the pipeline of ControlNet. E indicates encoder, SD is the Stable Diffusion, and c_f is the latent representation of the segmentation map Condition.

CHAPTER 2

Background Theory

2.1 Graph Convolution Network (GCN) and Cascaded Refinement Network (CRN)

A Graph Convolution Network (GCN) [6] works similarly to the traditional 2D Convolutional Neural Network. Given an input graph with vectors of dimension D_{in} for each node (objects) and edge (relations), the GCN computes new vectors of dimension D_{out} for the nodes and edges of the graph. Similar to CNN, the output is a function of the neighborhoods of the input, and the network Graph Convolution layer propagates information along the edges of the graph. A more detailed figure about the operations can be seen in Appendix A.2.

Cascaded Refinement Network (CRN) [3] is a neural network that generates images from semantic layouts. Unlike the recent works, it does not rely on adversarial training, but training with a regression loss. The network consists of Cascaded Refinement Modules which are made of Convolution and Upsampling layers. An interested reader could refer to the Supplementary Material of [8] for more details of the architecture.

2.2 Stable Diffusion Model and ControlNet

Stable Diffusion is a text-to-image latent diffusion model proposed by Rombach et al. [12]. A Latent diffusion model consists of three main parts Variational Autoencoder (VAE), U-Net, and a text-encoder (CLIP in this project). U-Net is a conditional model that gets input of time steps and noisy latent generated by a Variational encoder and predicts the noise using text embeddings as guidance. The detailed architecture can be seen in Appendix A.2 Figure A.3.

ControlNet recently proposed by Zhang et al. [18] is a neural network architecture that adds spatial conditioning controls to the diffusion models (Stable Diffusion in this project). The architecture uses the trainable copy of the U-net layers to learn specific conditional control. This architecture is of interest for small datasets such as scene graph datasets, due to its good learning ability with smaller datasets. In this project, I use a ControlNet that is pre-trained on Segmentation maps. A detailed architecture of ControlNet with Stable Diffusion can be seen in Appendix A.2 Figure **??**

2.3 Metric Scores

To evaluate the fidelity (quality) and diversity of the generated images I use three of the most commonly used metrics, Inception Score (IS) [13], Frechet Inception Distance (FID) [7] and Kernel Inception Distance (KID) [1]. These three scores are based on Inception v3 classifier, IS uses conditional and marginal probability, while FID and KID measure the distance between the real and generated images. KID is similar to FID however it is an unbiased estimator which makes it more suitable for smaller datasets, like in this project. The formulas of the scores can be found in Appendix A.3.

CHAPTER 3

Methods and Experiments

3.1 From Scene Graph to Segmentation Map

I use sg2im proposed by Johnson et al. [8] as a benchmark. The generator model consists of a Graph Convolution Network encoder and a Cascaded Refinement Network decoder, which can be seen in Appendix A.1. The scene graph is encoded as object vectors. Each vector is input to an object layout network consisting of a mask regression network and a box regression network. The layout network object layout and the sum of the object layouts is the scene layout. The architecture can be seen in appendix Figure A.2.

I use the same scene graph encoder for the ContolNet model, however some preprocessing is required so that the scene layout fits the input dimension of ControlNet. More precisely, each object layout is upsampled to the size of $(512, 512)$ using linear interpolation, and then the object layout is smoothed using Gaussian blur with a kernel size of 5. This kernel size and smoothing method were computationally efficient with high visual quality based on the pilot experiments with median blur, box blur, and kernel sizes of 3,5, and 7. I use OTSU thresholding for generating the segmentation maps.

3.2 Dataset

Visual Genome version 1.4 (VG) [9] which consists of 108,077 images annotated with scene graphs. Similar to sg2im [8], I ignore the small objects, and use images with between 3 and 30 objects and at least one relationship. In this project, I furthermore ignored the objects that are not present in ADE20k Dataset [19], leaving the test dataset of 1130 images with a total of 52 distinct objects with an average of six objects per image, ranging from 3 to 22 objects per image. The average of five relationships per image ranges from 1 to 23 relationships. The list of the objects and relationships of the dataset can be seen in Appendix A.4.

3.3 Settings and Implementation Details

I use the sg2im model trained on Visual Genome images with resolution $(64, 64)$. For further information, the reader is encouraged to refer to [8] and their GitHub repository.

3.3.1 Stable Diffusion Conditioned on Text

I use Stable Diffusion version 1-5 from Huggingface [10]. For optimal memory management I use *enable-model-cpu-offload()*, and to obtain a satisfactory trade-off between image quality and inference time, I use scheduler *UniPCMultistepScheduler* instead of the default scheduler *PNDMScheduler* as suggested by [17]. The rest of the inference parameters are set to default values, including guidance-scale set to 7.5. The negative prompt is set to default of empty string.

To generate a text input for the Stable Diffusion I merged all the relationships of the given scene graph in one sentence, separated by a comma. I also added the objects in the graph after the relationships. An example can be seen in Figure 3.1. CLIP text encoder is used to encode the prompt.

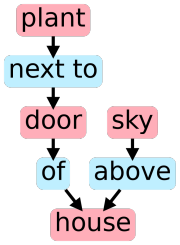


Figure 3.1: An Example of Scene Graph: The Stable Diffusion prompt is: "A photo of plant next to door, door of house, sky above house, plant, door, sky, house." The ControlNet prompt is: "A photo of plant, door, sky, house."

3.3.2 ControlNet

I use ControlNet model version 1.1 which is pre-trained on the ADE20k dataset [17]. Based on the analysis during pilot experiments, the model obtained significant improvement by using guess mode. This feature encourages the encoder to recognize the class of segmentation colors. The classifier guidance scale is set to 4 as advised by the documentations [17]. This combination is based on the qualitative performance during pilot experiments. The rest of the pipeline setting is the same as Stable Diffusion.

The prompt in the quantitative experiment is a sentence consisting of objects from scene graphs, seen in 3.1. To evaluate how capable ControlNet is in generating high-quality images independent of user prompt, I also qualitatively tested the model with the "default" prompt setting of "a photo". This way I evaluate qualitatively how capable the ControlNet encoder is in recognizing the objects of the input layout image. While it is interesting to investigate different levels of prompt settings, due to time constraints this project focuses on perfect prompt and default prompt settings.

3.3.3 Quantitative Experiments

I use a manual seed generator of 0 for both of the models. I use 30 iterations for inference, since based on the pilot experiments on values of $[20, 30, 50, 100]$ this value has a satisfactory visual quality while being computationally cheap. To have a fair comparison with sg2im, I decided to generate 1 image per scene graph. This is the same for both of the models.

The outputs of the diffusion models are down-sampled to $(299, 299)$, while outputs of sg2im are up-sampled to this value using nearest neighbor interpolation. The real images are also resized to this size to have the same size as the training data of the underlying Inception model. The inception score and KID are calculated across 10 splits of the test dataset, and there are randomly 100 samples taken from each split to calculate KID score.

3.3.4 Qualitative Experiments

I randomly chose 10 scene graphs from the VG test set and evaluated the quality of the generated images using different models visually. For this section, I perform inference in 50 iterations and both of the models generate a batch of 4 images per scene graph. Furthermore, to have a visually fair comparison with sg2im I downsample the diffusion models' output using Bicubic interpolation.

I furthermore visually evaluate the capability of the models to generate accurate images with recognizable objects and relationships from the scene graph.

To summarise, I have the following four setups for the experiments, point 3 is limited to qualitative experiment only:

1. text(CLIP) → Stable Diffusion
2. Scene Layout (sg2im encoder + post-process) + text(CLIP) → ControlNet
3. Scene Layout (sg2im encoder + post-process) + default-text(CLIP) → ControlNet
4. scene graph → sg2im

The code can be found on the link below:

https://github.com/Yeganeh1377/ImageGenerationFromSceneGraphsUsingControlNet/tree/main

CHAPTER 4

Results and Analysis

4.1 Quantitative Results and Analysis

The quantitative results seen on table 4.1, indicate that the diffusion models (ControlNet And Stable Diffusion) generate higher quality images with a higher level of fidelity than sg2im (benchmark) concerning FID and KID. Additionally, the increase in Inception Score (IS), it becomes evident that diffusion models generate images characterized by both superior quality and greater diversity compared to Sg2im.

At first glance one may conclude that ControlNet outperforms Stable Diffusion in generating images with higher fidelity, based on FID and KID. However, it should be noted that the difference between these two models is not significant as the confidence intervals of IS and KID overlap. Lastly, it should be noted that no certain conclusion can be drawn since the sample size of 1130 is very low.

Table 4.1: Quantitative Results Using 1130 Scene Graphs From the VG Test Split.

Method	**FID**	**IS**	**KID** ($\times 10^2$)
Real Images	_	20.2 ± 1.3	_
Sg2im	66.2	15.9 ± 0.9	1.8 ± 0.4
ControlNet	**31.9**	18.9 ± 2.1	$\mathbf{0.7 \pm 0.3}$
Stable Diffusion	39.9	$\mathbf{19.0 \pm 2.1}$	0.8 ± 0.2

4.2 Qualitative Results and Analysis

It can be seen in some of the randomly drawn generated images in Figure 4.1 that the images generated by diffusion models have higher quality and the objects are easier to distinguish compared to sg2im. Furthermore, ControlNet with default prompt is capable of distinguishing the objects in the input image and generates high-quality images. In some cases such as row 4, the low quality of the image generated with ControlNet using the default prompt can potentially be improved if the classifier-free guidance scale (CFG) is fine-tuned or as suggested in [18] using CFG Resolution Weighting instead. However, to my knowledge, this feature is not implemented yet.

It is furthermore interesting to note that the quality of the generated images using ControlNet with any of the prompt settings and sg2im depends greatly on the quality of the scene layout. For instance, these two models perform poorly for sparse scene layouts such as row 3.

Another interesting observation from row 2 and row 5 is that the models ControlNet and sg2im seem more capable of counting similar objects compared to Stable Diffusion (CLIP). This is also observed in the other generated images of the batch, seen in rows 7-9 of Figure A.8. This is potentially because the CLIP text-encoder is known to be incapable of counting [5]. However, ControlNet and sg2im are incapable of counting in the cases where the different instances of the same object category overlap in the segmentation map, seen in the second case of Figure A.8.

Figure 4.1: Five Scene Graphs and Generated Images Using Different Models: The second column is the input condition to ControlNet. The models generate high-quality images using relatively complex scene graphs with multiple objects and relations. The performance of sg2im and ControlNet depends on the quality of the scene layout. More examples can be seen in appendix A.7 and A.8.

I have furthermore performed a brief attempt to measure the number of recognizable scene graph objects and relations in the generated images. I have down-sampled the generated images from diffusion models using BICUBIC interpolation to ensure fair comparison to sg2im. For this part, I manually investigate 10 scene graphs. Each diffusion model generates 4 images per scene graph. I randomly select one image per each scene graph. The scene graphs and images can be seen in Figures 4.1 and A.7.
There are in total 54 objects and 51 relationships in the scene graphs. For easier comparison, I calculate the percentage of the scene graph objects and relationships that exist in the generated images by answering the question *How many of the objects/ relationships of the given scene graph are present in the generated image?*

While ControlNet and sg2im have slightly higher percentage of the recognizable objects and relationships, the difference is not significant. Hence, we cannot confirm that adding extra conditioning on structured scene graphs to the unstructured text, has in fact improved the accuracy of the image gen-

Table 4.2: The fraction of the recognizable objects and relationships across 10 scene layouts: ControlNet has the largest percentage of recognizable objects while sg2im has the largest percentage of the relationships. The poor performance of ControlNet with default prompt could potentially be improved by fine-tuning the hyer-parameters. No certain conclusion can be drawn from this analysis due to the very small sample size.

Method	**Objects** (%)	**Relationships** (%)
Sg2im	67	**63**
ControlNet	**69**	57
ControlNet(default)	54	29
Stable Diffusion	65	57

eration models. ControlNet with Default prompt performs poorly compared to other models. In some examples such as row 1 and row 5 of Figure A.7, the model in some difficult examples cannot recognize the class of the objects. The poor performance is because of the lack of recognition capability of model in sparse scene layouts.

However, it should be noted that no certain conclusion can be made from this analysis since the sample size is small and the results are biased toward the perception of one user. A user study with more samples and users are required for a certain conclusion.

CHAPTER 5

Discussion and Conclusion

5.1 Discussion and Conclusion

Scene layouts, including bounding boxes produced by sg2im [8] are not optimal and very coarse, which is the current bottleneck of the ControlNet system. This bottleneck could potentially be improved if the ControlNet is fine-tuned on VG dataset together with sg2im. Additionally, ControlNet inherits some of the limitations of the semantic maps. For instance, in the more complex case where there are multiple instances of the same category with overlapping regions, the model is incapable of counting the objects.

The current encoder pipeline has some bottlenecks when it comes to actions and non-spatial relations. Some simpler non-spatial relations such as ”house has door”, ” door of the house” and ”person riding horse” can be represented in the scene layout as the bounding box of the door is covered by the bounding box of the house or person on the horse. However, the more complex actions such as ”person eating food” cannot be represented well in the scene layout. A very interesting future could be the investigation of generation performance when the generation is directly conditioned on the embeddings rather than scene layouts.

To answer the first research question, it seems that ControlNet is superior to sg2im in generating high-quality, diverse images that contain recognizable objects. I believe that the performance of ControlNet could be further improved if (1) the model is fine-tuned on Visual Genome data, (2) The quality of the scene layouts is improved by using a better scene graph encoder. (3) Fine-tuning the hyper-parameters of the model such as classifier-free-guidance. Each of these could be an interesting idea for future work.

It is difficult to answer the second question since both of the models have similar performance quantitatively and visually. This is somewhat expected as the models have similar architectures. However, one may argue that ControlNet is easier to use as it is less dependent on prompt engineering. Lastly, it should be noted that the sample size is very limited for any certain conclusion.

Bibliography

[1] Mikołaj Bińkowski et al. "Demystifying MMD GANs." In: *International Conference on Learning Representations*. 2018. URL: https://openreview.net/forum?id=r1lUOzWCW.

[2] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. "Coco-stuff: Thing and stuff classes in context." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pages 1209–1218.

[3] Qifeng Chen and Vladlen Koltun. *Photographic Image Synthesis with Cascaded Refinement Networks*. 2017. arXiv: 1707.09405 [cs.CV].

[4] Azade Farshad et al. "Scenegenie: Scene graph guided diffusion models for image synthesis." In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pages 88–98.

[5] Frank Fundel. "Scene Graph Conditioning in Latent Diffusion." In: *arXiv preprint arXiv:2310.10338* (2023).

[6] C. Goller and A. Kuchler. "Learning task-dependent distributed representations by backpropagation through structure." In: *Proceedings of International Conference on Neural Networks (ICNN'96)*. Volume 1. 1996, 347–352 vol.1. DOI: 10.1109/ICNN.1996.548916.

[7] Martin Heusel et al. *GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium*. 2018. arXiv: 1706.08500 [cs.LG].

[8] 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*. 2018, pages 1219–1228.

[9] Ranjay Krishna et al. "Visual genome: Connecting language and vision using crowdsourced dense image annotations." In: *International journal of computer vision* 123 (2017), pages 32–73.

[10] Patrick von Platen et al. *Diffusers: State-of-the-art diffusion models*. https://github.com/huggingface/diffusers. 2022.

[11] Alec Radford et al. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. arXiv: 2103.00020 [cs.CV].

[12] Robin Rombach et al. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. arXiv: 2112.10752 [cs.CV].

[13] Tim Salimans et al. *Improved Techniques for Training GANs*. 2016. arXiv: 1606.03498 [cs.LG].

[14] Renato Sortino, Simone Palazzo, and Concetto Spampinato. *Transformer-based Image Generation from Scene Graphs*. 2023. arXiv: 2303.04634 [cs.CV].

[15] Steins. *Stable Diffusion Clearly Explained*. https://medium.com/@steinsfu/stable-diffusion-clearly-explained-ed008044e07e. 2023.

[16] Steins. *Stable Diffusion$_C$ontrolNetClearlyExplained*. https://medium.com/p/f86092b62c89#0bec. 2023.

[17] *Ultra fast ControlNet with Diffusers*. https://huggingface.co/blog/controlnet. Accessed: 2023-01-21.

[18] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. "Adding Conditional Control to Text-to-Image Diffusion Models." In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. October 2023, pages 3836–3847.

[19] Bolei Zhou et al. "Scene parsing through ade20k dataset." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pages 633–641.

APPENDIX A

Appendices

A.1 sg2im Model Architectures

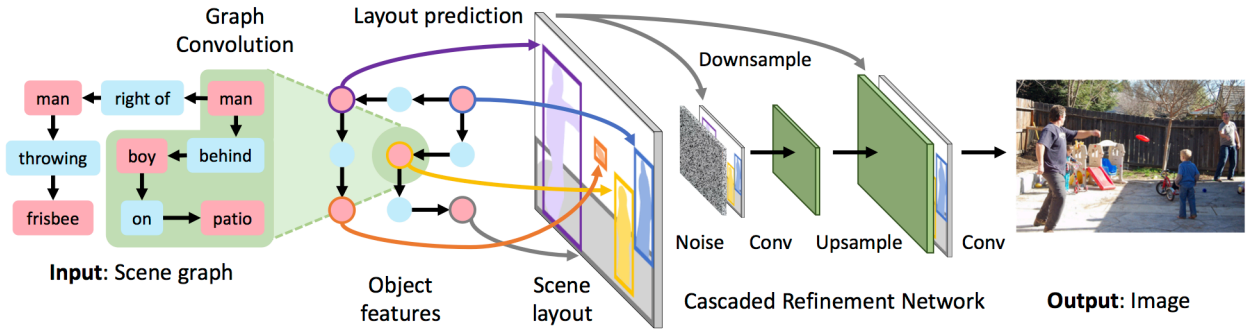


Figure A.1: The architecture of the Generation system of sg2im. The model is trained against two discriminators: patch-based image discriminator ensures the overall appearance of the generated image, and object discriminator ensures that the objects seem realistic. Figure taken from [8]

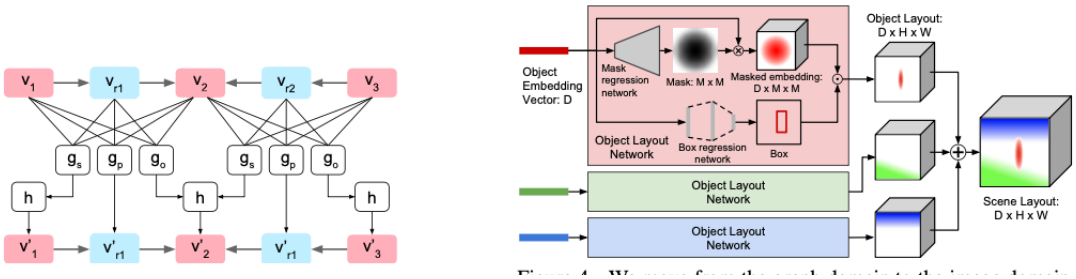


Figure A.2: Left: A layer of GCN. Right: The system of generating scene layout from object embedding vectors. Both of the figures are taken directly from [8]

.

A.2 Diffusion Models Architectures

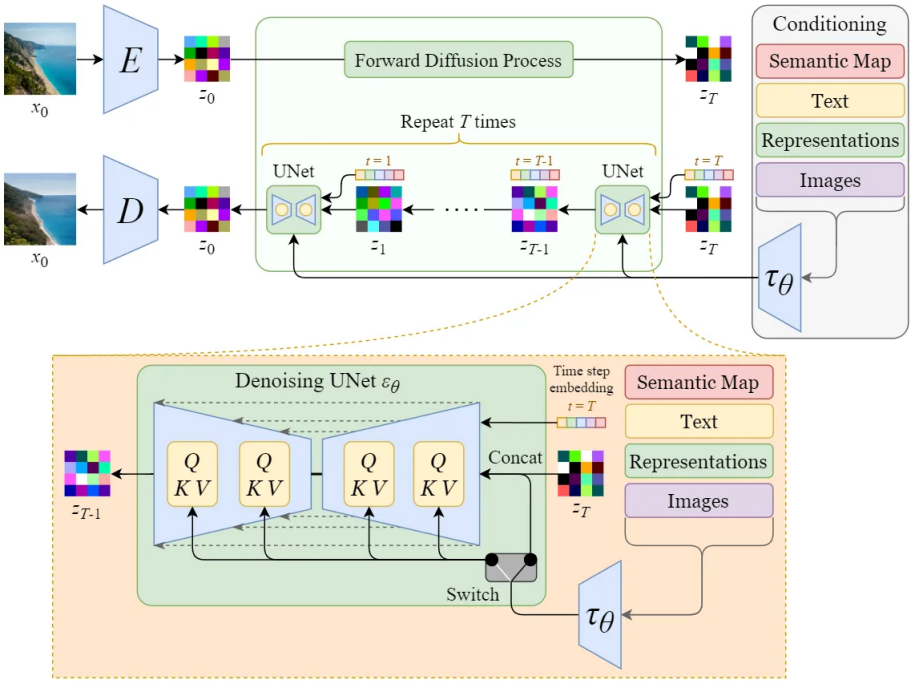


Figure A.3: Detailed Architecture of Stable Diffusion Model: τ is the language model CLIP in this project. The vector embeddings are mapped into U-Net via Attention layers called QKV, E and D are respectively encoder and decoder of the VAE model. In this project the condition is text. The Figure is taken from [15]

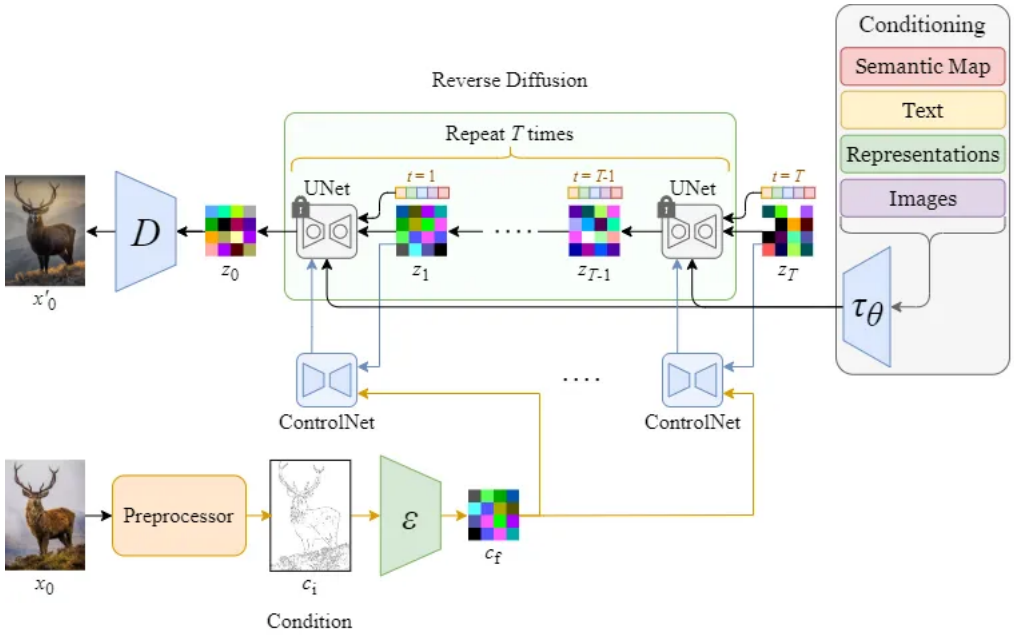


Figure A.4: Detailed Architecture of ControlNet with Stable Diffusion Model: ϵ is a small encoder which encodes the control image (segmentation map in the project to latent). In this project I directly give the segmentation map hence there is no pre-processing. The Figure is taken from [16]

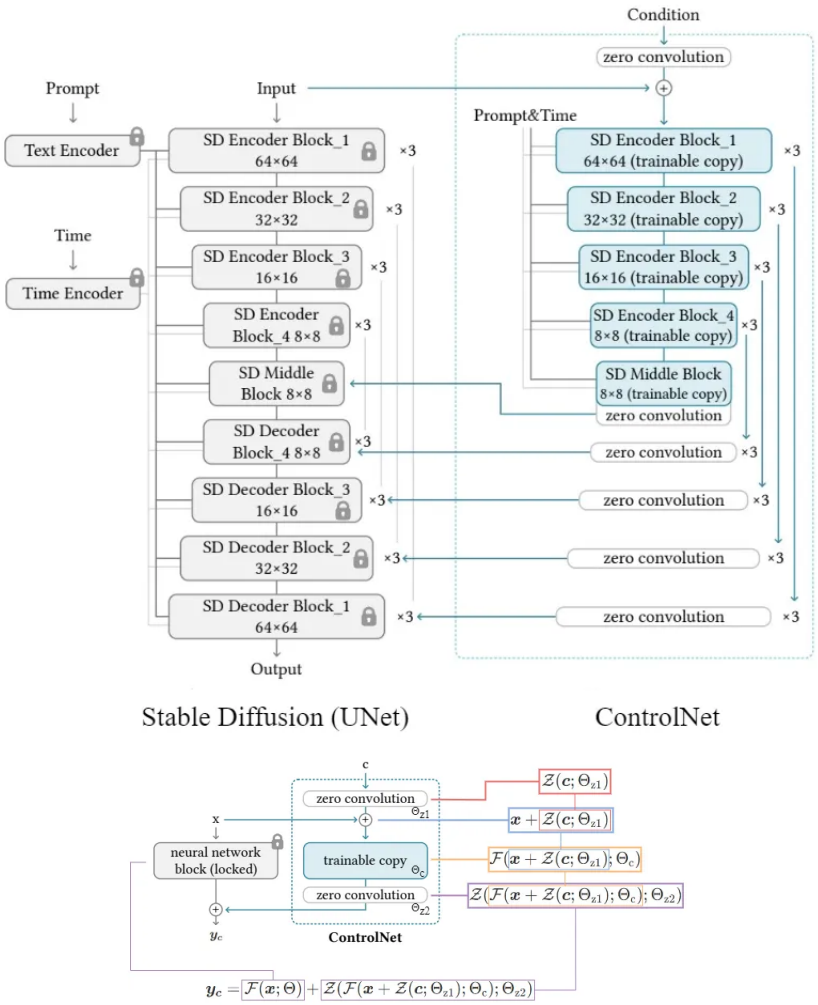


Figure A.5: Above is a detailed architecture of ControlNet. Below is visualization of ControlNet feedforward. x and y are deep features in the network, c is condition, $Z(.;.)$ is Zero convolution operation, while $F(.;.)$ is neural network block operation. Θ_{z1} and Θ_{z2} are respectively the parameters of the first and second zero convolution. Lastly, Θ_c is the parameters of the trainable copy. The Figures are taken from [16]

.

A.3 Formulas of The Metrices

$$IS = exp(\mathbb{E}_x KL(p(y|x)||p(y)))$$

$$\text{FID} = ||\mu - \mu_w||_2^2 + \text{tr}(\Sigma + \Sigma_w - 2(\Sigma^{1/2}\Sigma_w\Sigma^{1/2})^{1/2}).$$

$$KID = MMD(f_{real}, f_{fake})^2$$

Figure A.6: Formulas of the Metrics.

.

A.4 List of Objects and Relations

The following table contains the list of relationships:

in_image	has	covering	next to	above	belonging to	have	beside	behind
by	laying on	hanging on	eating	under	for	on side of	standing in	with
standing on	below	of	against	attached to	parked on	holding	on top of	carrying
at	on	wearing	in front of	looking at	wears	sitting in	near	over
sitting on	in	inside	walking on	along	made of	riding	covered in	around

The objects of the subset of Visual Genome are the following:

image	tree	person	wall	building	sky	grass	pole	light
car	table	water	door	fence	floor	chair	plate	road
sidewalk	flower	bag	field	glass	rock	bottle	boat	food
mirror	bench	clock	box	bus	shelf	pillow	plant	lamp
counter	house	flag	seat	book	ceiling	ball	truck	cabinet
hill	sand	sink	animal	mountain	railing	towel		

A.5 More Qualitative Results

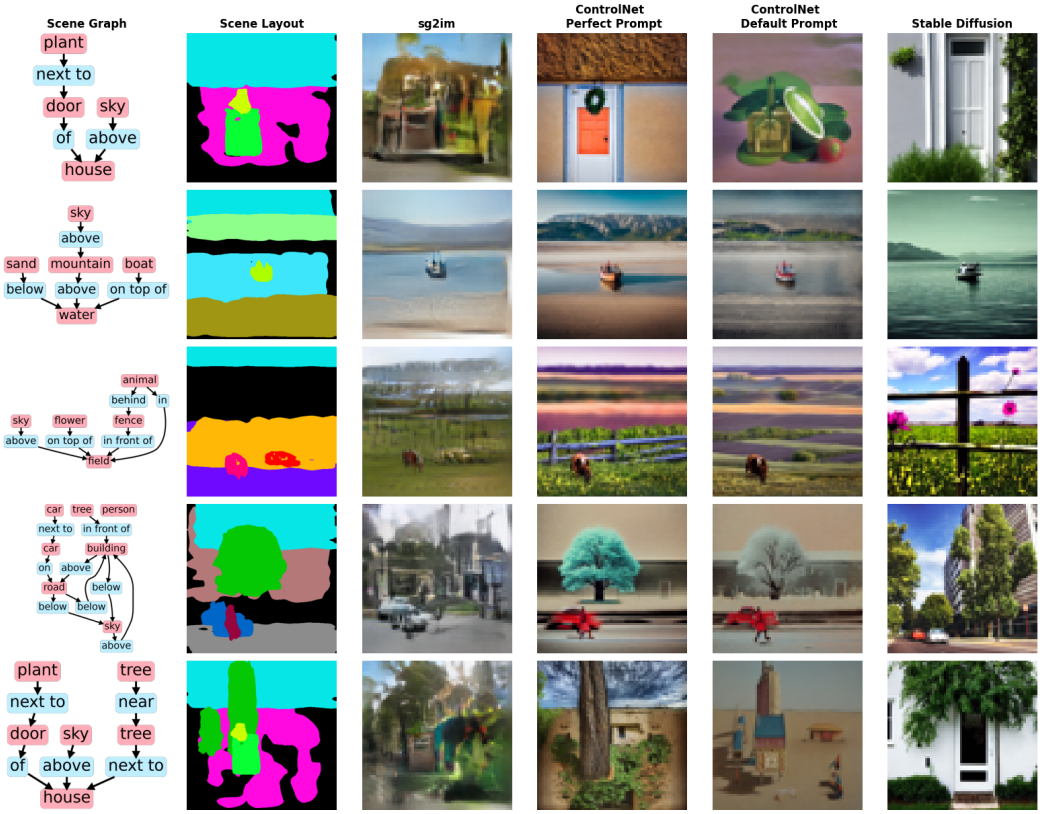
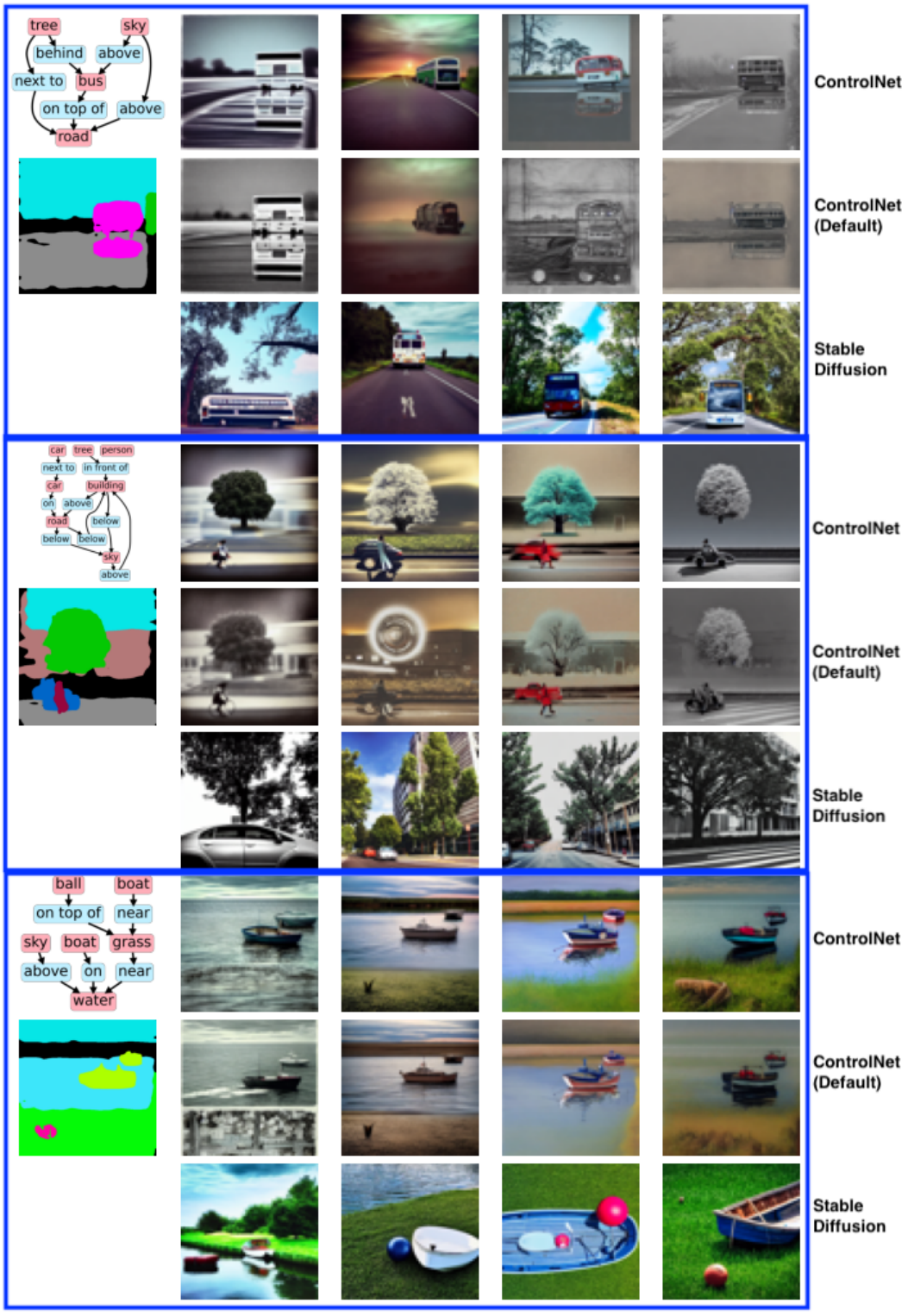


Figure A.7: The Rest of The Examples of the Generated Images


tree
sky
behind
above
next to
bus
on top of
above
road
ControlNet
ControlNet (Default)
Stable Diffusion
car
tree
person
next to
in front of
car
building
on
above
road
below
below
below
sky
above
ControlNet
ControlNet (Default)
Stable Diffusion
ball
boat
on top of
near
sky
boat
grass
above
on
near
water
ControlNet
ControlNet (Default)
Stable Diffusion

$\sqrt{17}$
$\Omega \int \delta e^{i\pi} = -1$
{2.7182818284}
ϑφερτυθιοπσδφγηξκλ
χ^2
$\sum$
!