Complex Scene Image Editing by Scene Graph Comprehension

Zhongping Zhang[1]
zpzhang@bu
Huiwen He[1]
huiwenhe@bu.edu
Bryan A. Plummer[1]
bplum@bu.edu
Zhenyu Liao[2]
zyliao@amazon.com
Huayan Wang[3]
wanghy514@gmail.com

[1] Boston University
MA, USA
[2] Amazon
CA, USA
[3] Kuaishou Technology
CA, USA

Abstract

Conditional diffusion models have demonstrated impressive performance on various tasks like text-guided semantic image editing. Prior work requires image regions to be identified manually by human users or use an object detector that only perform well for object-centric manipulations. For example, if an input image contains multiple objects with the same semantic meaning (such as a group of birds), object detectors may struggle to recognize and localize the target object, let alone accurately manipulate it. To address these challenges, we propose a two-stage method for achieving complex scene image editing by Scene Graph Comprehension (SGC-Net). In the first stage, we train a Region of Interest (RoI) prediction network that uses scene graphs and predict the locations of the target objects. Unlike object detection methods based solely on object category, our method can accurately recognize the target object by comprehending the objects and their semantic relationships within a complex scene. The second stage uses a conditional diffusion model to edit the image based on our RoI predictions. We evaluate the effectiveness of our approach on the CLEVR and Visual Genome datasets. We report an 8 point improvement in SSIM on CLEVR and our edited images were preferred by human users by 9-33% over prior work on Visual Genome, validating the effectiveness of our proposed method. Code is available at github.com/Zhongping-Zhang/SGC_Net.

1 Introduction

Text-to-image editing aims to modify the specific content of an image based on language descriptions. Prior work (*e.g.*, [2, 11, 22, 30, 32]) either edits the input image globally, where the entire image is subject to modification [11, 15], or localizes the Region of Interest (RoI[1]) according to either human-drawn [1, 22, 30] or detected bounding boxes [2]. These

© 2023. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

[1] We use RoI to indicate the region that is supposed to be edited in the image.

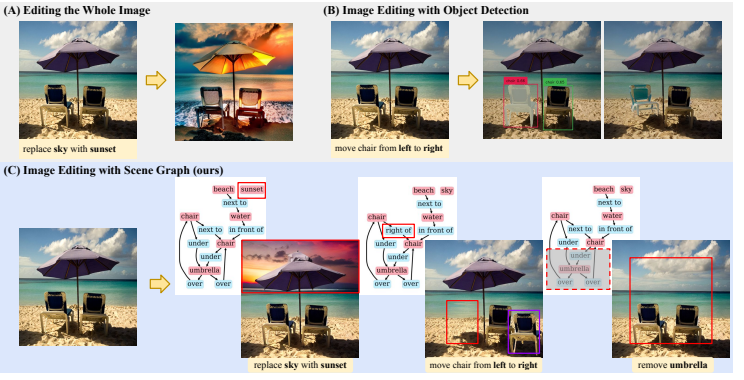

Figure 1: Prior work on text-to-image editing generally follow two approaches: (A), editing the entire image (*e.g.*, Imagic [11] or Dreambooth [31]), or (B), localizing the Region of Interest (RoI) through user-provided bounding boxes or object detection (*e.g.*, Grounded-SAM [12, 19]+Stable Diffusion [30]). In our work, shown in (C), we localize the RoI (outlined by the red bounding box) and predict the desired region (outlined by the purple bounding box) for the target object using a scene graph of an input image. This enables us to perform many editing operations in complex scenes, such as relationship change and object replacement, which were not supported by prior work.

methods have two major drawbacks. First, accurately localizing and moving a target object in complex scenes can be challenging due to inherent ambiguities of text. For example, in Figure 1(A), methods like Imagic [11] edits the entire image instead of the target object "sky." In Figure 1(B), Liu *et al.* [19] randomly selects a chair since they have the same semantic meaning. The second major drawback we observe is that most existing image inpainting methods are trained using free-form masks [21, 30, 37]. As our experiments will show, these methods may fail to add or replace the target object according to prompts, especially when the RoI is small. These methods fill in the masked regions by guessing at the background, instead of inserting the desired object according to the text prompt.

To address the aforementioned issues, we propose the Scene Graph Comprehension Network (SGC-Net) for complex scene image editing. In Figure 1(C), where the desired edit is "move the chair from left to right," the model must first localize the target object (the chair on the left side) given the text description. The model then must recognize the desired action of placing the chair to the right-hand side of the second chair. SGC-Net accomplishes this by converting the desired edits into changes in a scene graph representation of the image. Then, given the new scene graph, SGC-Net produces the edited image. This approach aids SGC-Net in more accurately identifying the target object despite textual ambiguities than prior work (*e.g.*, [12, 19, 22, 30]), as well as helps ensure that the edited object has the correct relationship to other objects in the scene.

More formally, given an image to be edited, we extract editing triplets from the given text prompt, *e.g.*, <target chair – right of – reference chair>, that we use to make changes to the scene graph representing the input image. Rather than relying solely on text descriptions that require a model to implicitly infer any changes to the spatial information in an image, our model explicitly predicts the desired location of target object(s) based on their relationship to other objects encoded in the scene graph. For example, for the triplet <target chair – right of – reference chair>, the first stage of our model uses an RoI prediction module to identify that the desired region for the target chair should be on the right side of the reference chair

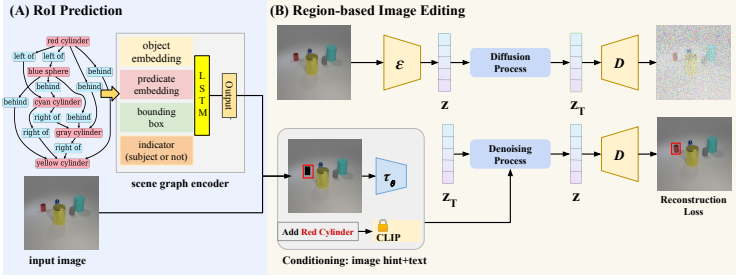

Figure 2: **SGC-Net overview.** Our approach consists of two sequential stages: (A) RoI prediction, which localizes the RoI based on a modified scene graph. Specifically, a scene graph encoder takes the modified scene graph as input and outputs a bounding box. (B) Region-based image editing: During inference, we take the image masked by predicted bounding boxes as input and outputs the modified image. During training, we randomly mask out objects to simulate the output of our RoI prediction module, and train our model to reconstruct the original image. Thus, our model does not require image pairs before and after editing.

(illustrated in Figure 2(A)). Then, in the second stage SGC-Net edits the image conditioned on the text prompt and RoI predictions from the first stage (shown in Figure 2(B)).

In summary, the contributions of this paper are:

- We propose a Scene Graph Comprehension Network (SGC-Net) that performs text-to-image editing using scene graphs, reducing manual effort and alleviating the issues caused by the ambiguity of text. Compared to methods that require pre-defined masks or bounding boxes as input [22, 30], our model uses the semantic meanings of objects and their relationships in an image to accurately predict desired RoI for the target object.
- We propose a region-based image editing approach built on Stable Diffusion [30] that can achieve different image editing tasks without paired training images. As we will show, our approach can even remove or add objects with small bounding boxes.
- Our experiments demonstrate SGC-Net outperforms the state-of-the-art. *E.g.*, SGC-Net provides an 8 point boost in SSIM (RoI) on CLEVR [10] and human users preferred SGC-Net image edits by 9-33% on Visual Genome [13] over the baselines [4, 22, 30].
- We perform experiments on in-the-wild[2] image editing, validating the generalization and flexibility of our method.

2 Related Work

Text-to-Image Synthesis & Editing. Early studies in text-to-image synthesis and editing often relied on conditional GANs [6, 15, 18, 35, 39], but often were limited to generating low-resolution images due to high GPU memory requirements and scalability limitations of these methods. More recent methods (*e.g.*, [27, 28, 30, 32, 41]) have focused on training conditional diffusion models for text-to-image generation on large scale datasets (*e.g.*, LAION-400M [33]). Building on these text-to-image generation frameworks, different editing approaches [2, 7, 11, 22, 31] have been proposed to support image editing tasks such as attribute manipulation and image inpainting. These methods can be roughly divided into two major categories. First, methods that perform global image edits based on text prompts,

[2]Following [42], we refer to images sourced from the real world as "in-the-wild", as opposed to image datasets collected by humans.

such as Dreambooth [31], Imagic [11], and Text Inversion [7]. They treat the entire image as the target object and manipulate the input image accordingly. The second category of methods [2, 22, 30, 43] use region-based editing that only modifies the Region of Interest (RoI). These methods obtain the target region using either user-provided masks [22, 30] or those from an object detector [2, 19]. While these methods work well for images with few objects and simple semantic relationships, as we have discussed, accurately locating and manipulating target objects in complex scenes (*e.g.*, those with multiple object that have similar semantic meanings) can be challenging. To address this, our SGC-Net utilizes a scene graph encoder to accurately localize the RoI by taking advantage of object relationships and predict the desired region for the target object without requiring user-provided masks.

Scene Graphs and Visual Relationship Detection. Scene graphs [9] describe the objects, attributes of objects, and relationships between objects in an image, and methods that generate them can be divided into two categories: Convolutional Neural Network (CNN)-based methods [16, 17, 26, 36] and Recurrent Neural Networks (RNN)-based methods [8, 34, 38]. At a high level, scene graph construction combines object/entity detection [25, 29] and detecting their visual relationships [3, 20, 24]. Our paper also is related to tasks where they aim to perform entity localization according to visual relationships (*e.g.*, [14, 24]). However, unlike these tasks, in our setting the target object is not always visible in the image. Therefore, we propose an RNN-based method to automatically predict a plausible position for the target object according to the existing information from the image and `<subject-predicate-object>` triplets.

3 SGC-Net: Scene Graph Comprehension Network for image editing

Given an input image x, we define the entities in x as $O = \{o_1, ..., o_n\}$, the corresponding bounding boxes as $B = \{b_1, ..., b_n\}$, and the relationships between entities as $R = \{r_1, ..., r_m\}$. Our task is to perform semantic manipulation on x based on a scene graph G that consists of $\{O, B, R\}$, where the scene graph can be directly modified by users or modified according to the user-provided text prompts. To manipulate the image x according to the scene graph G, our SGC-Net uses a two step process. First, Section 3.1 describes our Region-of-Interest (RoI) prediction module that identifies the target manipulation predicting the desired regions for the target object with scene graph information G. Second, Section 3.2 presents our region-based image editing model, which aims to understand and implement various image editing operations. Finally, Section 3.3 describes the editing operations and their corresponding modifications on scene graph. Figure 2 provides an overview of our approach.

3.1 Region of Interest (RoI) Prediction

As discussed in the Introduction, correctly localizing and predicting the desired regions for the target object is critical for complex scene image editing. To address the text ambiguity issues in methods based on object detection [2, 19], we propose a RoI prediction module based on scene graph to facilitate various image editing tasks. This module is especially useful for tasks like semantic relationship change and object addition since they explicitly identify the location to place the target object. This module enables SGC-Net to accurately localize the target object and predict where to place it, instead of randomly selecting one

object that satisfies the semantic meaning as the target object. Since the target object may have relationships with multiple objects in a scene, we purpose an LSTM-based model that encodes all the triplets of the modified scene graph.

Given a modified scene graph $\widetilde{G}$ and the target object o with triplets $\mathbf{y} = \{y_1,...,y_T\}$. For $t \in \{1,...,T\}$, y_t is a triplet in $< s-p-o >$ format, where $s,o \in O$ and $p \in R$. For each triplet y_t, we devise two embedding layers that obtain the object embedding V_s, V_o and predicate embedding V_p separately. We also introduce a binary indicator I to indicate whether the target object is subject or object in y_t. Suppose the reference object corresponds to "subject" and the target object corresponds to "object," we also consider the position of reference object b_s as part of the input to our model. We concatenate these features and encode them with an LSTM. Specifically, it can be expressed as:

$$m_t = concat\{V_s, V_o, V_p, b_s, I\}; \quad h_t = LSTM(m_t, h_{t-1}) \tag{1}$$

where h_t is the hidden state of LSTM at triplet y_t. We apply an Multilayer Perceptron (MLP) on h_T to predict the four coordinates of RoI. Since the image size is normalized, we crop the final output to range(0∼1) and train the model using Mean Squared Error. Ss shown in Figure 2, once we obtain the predicted RoI, we generate a masked image x_{hint}, where the RoI is masked out and given as input to the image editing module described in the next section.

3.2 Region-based Image Editing

After obtaining the predicted bounding boxes from our RoI prediction module described in Section 3.1, a straightforward approach to image editing is to apply an image inpainting method, such as GLIDE [22] or Stable Diffusion [30], directly to the masked image x_{hint}. However, as our experiments will show, these approach struggles to accurately add or remove objects in complex scenes, particularly when the bounding boxes are small, or distinguish between operations like addition and removal. For example, when asked to remove an object these methods tend to add the object to the modified region instead. This is due, in part, to the fact that many models (*e.g.*, [30, 32]) were pretrained on datasets like LAION-400M [33] that contain generic image captions that likely do not contain editing-specific language like "*remove* a car" or "*add* a car." Thus, during training we generate a text prompt using a template for each editing operation we support, and then train our region-based image editing module conditioned on our generated prompts. Additionally, rather than directly inpaint x_{hint}, we provide it as an additional control to our diffusion process. This enables SGC-Net to decide what parts are important to keep and what may be safely ignored.

More formally, our region-based image editing approach uses Stable Diffusion [30] to reconstruct the original image x from the masked image x_{hint} according to user's input. During training x_{hint} can be obtained by randomly removing objects from x, thereby eliminating the requirement for paired image data before and after editing. Specifically, given the image data x_0 and a randomly generated noisy image x_T, Stable Diffusion [30] can be considered as a series of equally-weighted denoising autoencoders $\varepsilon_\theta(x_t, t)$. Here T represents the length of the Markov Chain in the diffusion process, and t denotes the time step, ranging from 1 to T. For unconditional generation, the denoising autoencoders are trained to predict the corresponding noise ε with an objective function:

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(I),\varepsilon\sim\mathcal{N}(0,1),t}\left[||\varepsilon - \varepsilon_\theta(z_t, t)||_2^2\right], \tag{2}$$

where $\mathcal{E}$ is the pretrained encoder of VQGAN [8] to encode image x_t to latent features z_t, and vice versa. For conditional generation, the denoising autoencoders ε take $\tau_\theta(y)$ as additional input and the Conditional Latent Diffusion Model (CLDM) is optimized by

$$L_{CLDM} := \mathbb{E}_{\mathcal{E}(I),y,\varepsilon\sim\mathcal{N}(0,1),t}\left[||\varepsilon - \varepsilon_\theta(z_t, t, \tau_\theta(y))||_2^2\right], \tag{3}$$

where τ_θ and ε_θ are optimized jointly. As discussed earlier, during training our text prompts are created from a template that include information about the type of editing operation being performed. In addition, to perform editing operations around the bounding box (*e.g.*, add shadows to the red cylinder in Figure 2), we use x_{hint}, which is encoded by ControlNet [41], as an image hint in our conditions, rather than just as input to the denoising autoencoders $\varepsilon_\theta(x_t, t)$ like GLIDE [22] or Stable Diffusion [30]. This allows our model to perform various editing tasks both within and around the RoI, resulting in more visually natural outputs.

3.3 Image Editing Operations

Following Dhamo et al. [4], we perform four image manipulation tasks: object addition, object replacement, relationship change, and object removal[3]. These manipulations are reflected on the nodes and edges of scene graphs. Take the red cylinder in Figure 2 as an example, (1) Object Addition: adding a new object (node) and its corresponding relationships (edges) with other objects on the scene graph; (2) Object Replacement: replacing the node which represents `<red cylinder>` to another object; (3) Relationship Change: changing the spatial relationships of red cylinder (edge) from `<red cylinder-left of-blue sphere>` to `<red cylinder-right of-blue sphere>`. Other relationships of the red cylinder can be changed similarly; (4) Object Removal: deleting the node of `<red cylinder>` and its corresponding edges.

4 Experiments

Datasets. We evaluate the performance of our model on CLEVR [10] and Visual Genome [13]. CLEVR is a synthetic dataset that contains ground truth pairs for image editing. We use the test set provided by Dhamo et al. [4] for a fair comparison to prior work. However, Visual Genome, lacks image pairs before and after editing. Thus, we evaluate performance using human judgements to estimate the correctness of manipulation.

Metrics. Following Dhamo et al. [4], we report mean absolute error (MAE) and structural similarity index measure (SSIM) of RoI as the evalution metrics. The RoI is defined as the area of image that should be modified according to user's input, which can directly reflect the accuracy of the manipulation.

Baselines. We choose SIMSG [4], GLIDE [22], and Stable Diffusion (SDM) v2 [30] as our baselines for image editing. Since GLIDE and SDM require user-provided masks as input, we use the bounding boxes predicted by our RoI prediction module as input to GLIDE and SDM on CLEVR. For Visual Genome, we use Grounded-SAM [12, 19] to automatically detect the segmentation masks for GLIDE and SDM[4].

[3]Following Dhamo et al. [4], we perform all four operations on CLEVR [10], and three operations on Visual Genome [13]: object replacement, relationship change, and object removal.

[4]We refrain from using Grounded-SAM on CLEVR since its images always consist of multiple semantically similar objects, making it difficult for Grounded-SAM to accurately localize the target object.

Method	Object Removal		Object Replacement		Object Addition		Relationship Change	
	MAE ↓	SSIM ↑	MAE ↓	SSIM ↑	MAE ↓	SSIM ↑	MAE ↓	SSIM ↑
GLIDE [22]+RoI	39.51	62.91	34.59	55.57	40.25	59.04	-	-
SDM v2 [30]+RoI	31.48	72.65	36.49	54.48	44.39	57.61	-	-
SIMSG [4]	30.38	85.89	33.76	67.41	44.58	**65.80**	33.31	85.95
SGC-Net (ours)	**9.90**	**93.33**	**27.64**	**68.05**	**34.18**	64.78	**11.40**	**92.32**

Table 1: Quantitative results for editing 256×256 images on CLEVR. Omitted results indicate a method that cannot support a task. Since GLIDE and SDM requires user-provided inpainting mask, we use the predicted bounding boxes by our method as input, which is denoted by "RoI". SGC-Net notably outperforms baselines, especially for the object removal and relationship change tasks. See Section 4.1 for discussion.

Implementation Details. SGC-Net has two stages: the RoI prediction and region-based image editing. We train the two stages separately. To train the RoI prediction module, we extract modified scene graphs as input and the bounding boxes of target objects as output. We set the maximum number of the modified triplets of each image to 5. The mean absolute error (MAE) on the validation set of CLEVR is 12.05±0.87 (computed over 5 runs), with images of size 256×256. The MAE on the Visual Genome dataset is 77.84±1.25 (computed over 5 runs), with images of size 512×512. For the region-based editing model, we simulate the output x_{hint} of our RoI prediction model by randomly masking out objects. We use x_{hint} as input and the original image x as output. Finally, we adopt a compositional scene representation mechanism following Zhang et al. [43] to preserve the original content of unmodified regions. To eliminate obvious artificial effects in the boundary area, we apply Possion Blending [23] to combine the input image and the modified regions.

4.1 Results on CLEVR

We present quantitative and qualitative results on 256×256 images[5] in Table 1 and Figure 3, respectively. We draw three major conclusions. First, text-to-image inpainting methods [22, 30] pretrained on massive datasets do not perform well on our image editing tasks. As discussed in Section 3.2, they tend to introduce unexpected objects to repair the missing regions. This shown by the significant performance gains on object removal reported in Table 1 and also illustrated in the removal example in Figure 3, where GLIDE uses a distorted blue cube to fill the missing regions and SDM draws the yellow cylinder from the right-hand side to inpaint the masked region. Our second conclusion is that text-driven localization methods struggle to accurately localize the RoI and desired regions. *E.g.*, there are two grey cylinders and two purple spheres for image addition example in Figure 3. Even a perfect object detection method only has a 25% chance of correctly localize the target objects, let alone understand the semantic relationships between them. Finally, our third observation notes that although SIMSG achieves comparable SSIM scores with our method on object replacement and addition, the MAE loss of SIMSG is much higher than ours, which means some attributes like brightness may be modified by SIMSG (*e.g.*, as shown in Figure 3).

[5] We apply super-resolution method USRNet [40] on the output of SIMSG (128×128 on CLEVR and 64×64 on Visual Genome) to avoid out-of-memory issues on a 48GB GPU.

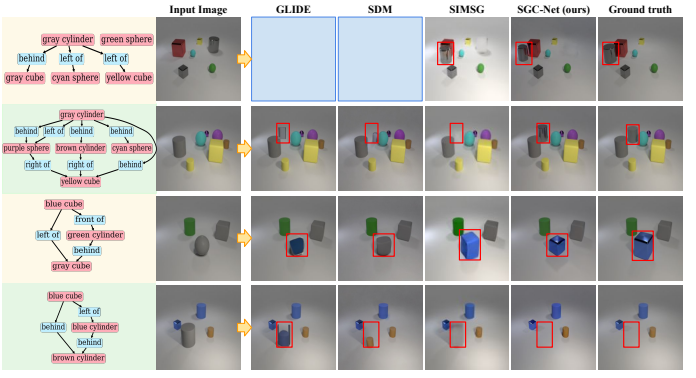

Figure 3: Qualitative examples for editing 256×256 images on CLEVR. Tasks from top to bottom: semantic relationship change, object addition, object replacement, and object removal. Blank image means the corresponding approach is not capable of this task. We outline the RoI by red bounding boxes. We observe that SGC-Net can accurately predict the RoI and edit objects in complex scenes. See Section 4.1 for discussion.

	Object Removal	Object Replacement	Relationship Change
GLIDE [22]	23.3%	28.8%	-
Stable Diffusion v2 [30]	21.1%	32.2%	-
SIMSG [4]	22.2%	24.4%	15.6%
SGC-Net(ours)	**50.0%**	**40.0%**	**48.9%**

Table 2: User judgments on the correctness of an image manipulation on Visual Genome. Empty values indicate the approach is not capable of this task. SGC-Net outperforms baselines in image manipulation accuracy, especially for object removal and relationship change, aligning with our observation on CLEVR results. See Section 4.2 for discussion.

4.2 Results on Visual Genome

As with experiments on CLEVR, we apply GLIDE and SDM on object removal and object replacement tasks, and SIMSG on all editing tasks. For each image editing task, we randomly selected 30 images generated by each baseline. This resulted in 120 images for object removal and replacement, and 60 images for semantic relationship change. Each image was annotated three times by AMT workers where they were asked to judge whether the image is correctly manipulated according to the input guidance. In Table 2, we report that SGC-Net significantly outperforms baselines, especially for object removal and relationship change tasks, which is consistent with our observation on CLEVR.

We present qualitative examples in Figure 4. Figure 4(A) shows that our RoI prediction module outputs reasonable desired regions for the target objects when changing its relationship to other objects in the scene, such as "person beside horse" and "bird on water". In contrast, methods such as GLIDE or SDM do not understand the semantic relationship change between objects and cannot edit the image accordingly. Figure 4(B) presents object replacement examples, where SGC-Net outperforms both the scene-graph driven approach SIMSG and the text-driven approaches GLIDE and BDM. For instance, GLIDE often guesses at the background without properly inserting the object into the masked area. Figure 4(C) shows

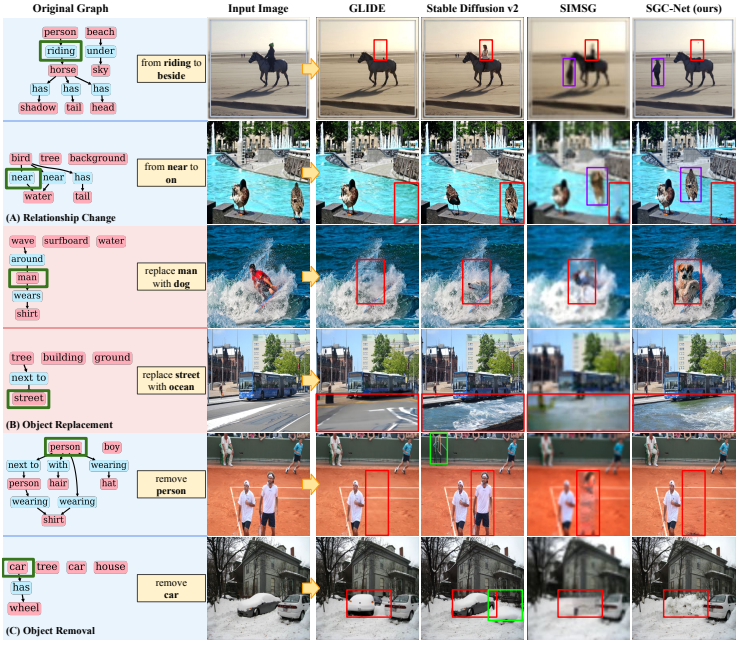

Figure 4: Qualitative examples for editing 512×512 images on Visual Genome. The original region of the target object, desired region for the target object are outlined by red bounding box and purple bounding box respectively. We simplify the scene graphs for better visualiation. SGC-Net not only predicts desired regions for the target objects but also edits these regions based on user modifications better than prior work. See Section 4.2 for discussion.

object removal examples. To demonstrate the effectiveness of our RoI prediction module, we apply the red bounding boxes predicted by our model as input to GLIDE and the green bounding boxes predicted by Grounded-SAM [12, 19] as input to SDM. It can be observed that without the input masks predicted by our model, SDM failed to accurately localize the target object due to the presence of multiple "cars" and "people" in the images. In contrast, GLIDE correctly localized the target object according to the bounding boxes provided by our method. However, as discussed in Section 3.2, methods like GLIDE may introduce unexpected objects to the image, as seen in the second object removal example where GLIDE introduces a new car covered by snow to inpaint the target region. Our region-based editing model, on the other hand, does not suffer from this issue due to its training strategy.

4.3 In-the-wild Text-to-image Editing

Our experiments have demonstrated that incorporating scene graph information can significantly improve the accuracy of image editing compared to using text alone. While modifying the scene graph incurs additional overhead compared to editing text only, we demonstrate that our model can be easily adapted to text commands in this section. We show diverse generated results in Figure 5, where our model accurately edits the target object according to the `<subject-predicate-object>` triplets that are converted from text commands. For instance, the cat has been moved from the sofa to the floor, and the spoon has been moved from inside the plate to beside the plate.

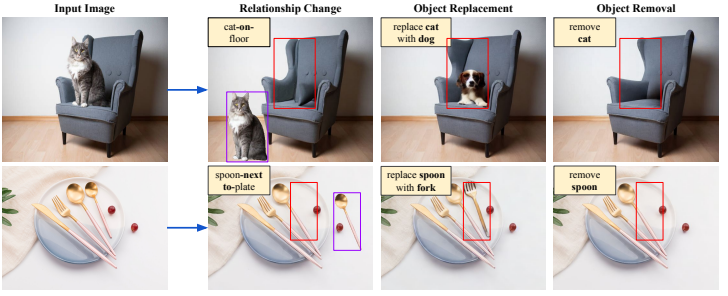

Figure 5: In-the-wild text-to-image editing results. SGC-Net can be adapted to text prompts with localization methods (*e.g.*, Grounded-SAM [17, 19]). See Section 4.3 for discussion.

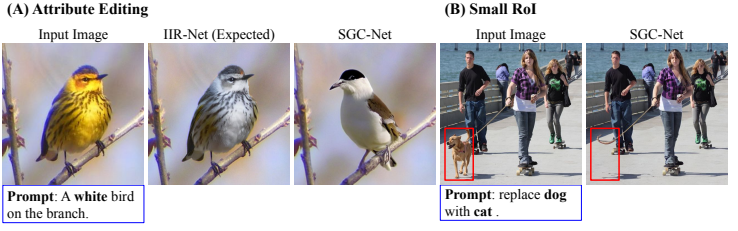

Figure 6: Limitations of SGC-Net. (A) Attribute editing: SGC-Net lacks the ability to modify object attributes. (B) Inserting objects in small RoI: While SGC-Net can effectively alleviate the background guessing issue, we still observe some failure cases on Visual Genome when the RoI is small. See Section 4.4 for discussion.

4.4 Limitations and Future Work

We find SGC-Net can edit objects in complex scenes, but struggles to edit object attributes. *E.g.*, as shown in Figure 6(A), we could replace yellow bird with a white bird, but cannot produce the same bird with a different attribute. Thus, exploring ways to integrate attribute editing methods(*e.g.*, Imagic [11] or IIR-Net [44]), to generate objects with different attributes is a good direction for future work. Additionally, while our method effectively alleviates the issue of inserting objects into small regions, we still observe some failure cases where it may struggle to insert the objects, particularly in some cases on Visual Genome shown in Figure 6(B). Based on this finding, collecting more training images containing small objects may further improve the performance on object editing for small RoIs.

5 Conclusion

In this paper, we propose SGC-Net, a semantic image editing method that uses scene graph information to achieve complex scene image editing. SGC-Net consists of two stages: an RNN-based RoI prediction module that identifies regions to edit the target objects, and a region-based image editing module that supports many editing tasks such as object replacement, removal, and relationship change. We demonstrate that SGC-Net outperforms the state-of-the-art in both qualitative and quantitative evaluations on CLEVR and Visual Genome. For example, SGC-Net achieves around an 8-point gain in SSIM on CLEVR and a 9-33% boost in user evaluation accuracy on Visual Genome. Furthermore, experiments on in-the-wild image editing demonstrate SGC-Net's ability to generalize to practical settings.

Acknowledgements. This material is based upon work supported, in part, by DARPA under agreement number HR00112020054. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the supporting agencies.

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022.

[2] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV*, pages 707–723. Springer, 2022.

[3] Bo Dai, Yuqi Zhang, and Dahua Lin. Detecting visual relationships with deep relational networks. In *Proceedings of the IEEE conference on computer vision and Pattern recognition*, pages 3076–3086, 2017.

[4] Helisa Dhamo, Azade Farshad, Iro Laina, Nassir Navab, Gregory D Hager, Federico Tombari, and Christian Rupprecht. Semantic image manipulation using scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5213–5222, 2020.

[5] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[6] Wan-Cyuan Fan, Cheng-Fu Yang, Chiao-An Yang, and Yu-Chiang Frank Wang. Target-free text-guided image manipulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 588–596, 2023.

[7] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

[8] Roei Herzig, Moshiko Raboh, Gal Chechik, Jonathan Berant, and Amir Globerson. Mapping images to scene graphs with permutation-invariant structured prediction. *Advances in Neural Information Processing Systems*, 31, 2018.

[9] 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.

[10] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910, 2017.

[11] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Conference on Computer Vision and Pattern Recognition 2023*, 2023.

[12] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

[13] 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.

[14] Ranjay Krishna, Ines Chami, Michael Bernstein, and Li Fei-Fei. Referring relationships. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6867–6876, 2018.

[15] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip HS Torr. Manigan: Text-guided image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7880–7889, 2020.

[16] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1261–1270, 2017.

[17] Yikang Li, Wanli Ouyang, Bolei Zhou, Jianping Shi, Chao Zhang, and Xiaogang Wang. Factorizable net: an efficient subgraph-based framework for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 335–351, 2018.

[18] Xiaodan Liang, Hao Zhang, Liang Lin, and Eric Xing. Generative semantic manipulation with mask-contrasting gan. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 558–573, 2018.

[19] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[20] 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. Springer, 2016.

[21] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. Edgeconnect: Structure guided image inpainting using edge prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[22] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning (ICML)*, 2022.

[23] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *ACM SIGGRAPH*, pages 313–318, 2003.

[24] Bryan A Plummer, Arun Mallya, Christopher M Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. Phrase localization and visual relationship detection with comprehensive image-language cues. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1928–1937, 2017.

[25] Bryan Allen Plummer, Kevin Shih, Yichen Li, Ke Xu, Svetlana Lazebnik, Stan Sclaroff, and Kate Saenko. Revisiting image-language networks for open-ended phrase detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[26] Mengshi Qi, Weijian Li, Zhengyuan Yang, Yunhong Wang, and Jiebo Luo. Attentive relational networks for mapping images to scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3957–3966, 2019.

[27] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[28] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[31] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022.

[32] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[33] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[34] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5410–5419, 2017.

[35] 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.

[36] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 670–685, 2018.

[37] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4471–4480, 2019.

[38] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5831–5840, 2018.

[39] 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.

[40] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3217–3226, 2020.

[41] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023.

[42] Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris Metaxas, and Jian Ren. Sine: Single image editing with text-to-image diffusion models. *arXiv preprint arXiv:2212.04489*, 2022.

[43] Zhongping Zhang, Huiwen He, Bryan A Plummer, Zhenyu Liao, and Huayan Wang. Semantic image manipulation with background-guided internal learning. *arXiv preprint arXiv:2203.12849*, 2022.

[44] Zhongping Zhang, Jian Zheng, Jacob Zhiyuan Fang, and Bryan A Plummer. Text-to-image editing by image information removal. *arXiv preprint arXiv:2305.17489*, 2023.