Generating Open-World & Multi-Hierarchy Scene Graphs for Human-instructed Manipulation Tasks via Foundation Models

Sandeep S. Zachariah*, Aman Tambi*, Moksh Malhotra, P. V. M. Rao and Rohan Paul
Indian Institute of Technology Delhi, India

***Abstract*— For generating viable multi-step plans in robotics, it is necessary to have a representation scheme for scenes that is both open-set and structured in a way that facilitates local updates when the scene changes. We propose a method for generating multi-hierarchical scene graphs in a zero-shot manner using foundation models, which can support downstream planning tasks. We demonstrate that our method yields superior results compared to previous works in both open-world object detection and relation extraction, even without any priors. Moreover, we illustrate how the multi-hierarchical nature of the scene graph aids the planner in devising feasible plans for tasks necessitating reasoning over the spatial arrangements and object category abstractions. Project web page: https://reail-iitdelhi.github.io/scenegraph.github.io/**

I. Introduction

Consider a robot asked to *"bring me all the fruits"* or *"put the book which has spectacles on top of it on the rack"*. Following such instructions requires the robot to possess a grounded semantic understanding of its environment in terms of which objects are present, their metric location/extent, how objects are related to each other (inside, left of, behind, supported by etc.). Further, such a representation must be scalable, allowing updation as the robot takes actions to affect its environment. This paper concerns enabling a robot to generate a *scene graph* of its environment in an open world setting i.e., without *a-priori* knowledge of which objects that the robot may encounter. Such a scene graph should model objects as well as a multitude of semantic relations to support diverse instructions that the robot may receive in future. Finally, the representation must support rapid reconstruction as the robot manipulates objects as intended by the instruction.

Traditional approaches use supervised learning methods to infer metric-semantic graph representations for a scene (e.g., [7]). Such approaches have shown significant successes in accurate metric modeling of the space and supporting semantic interactions. However, their reliance on a predefined set of objects limits their use in open-world settings. Recent works leverage common sense knowledge embedded in foundation models for open-vocabulary or zero-shot induction of semantic properties for a scene that a robot may be in. We build on such efforts (e.g., [6]) and evaluate SOTA models for zero-shot scene graph construction for robot instruction following. Our experiments revealed that direct use of VQA/VLM models for this task result in sparse graph with a large number of missing or hallucinated objects for practical scenes and limited ability to decode relations required for instruction following. Further, latency is high due to server access required for most large scale models.

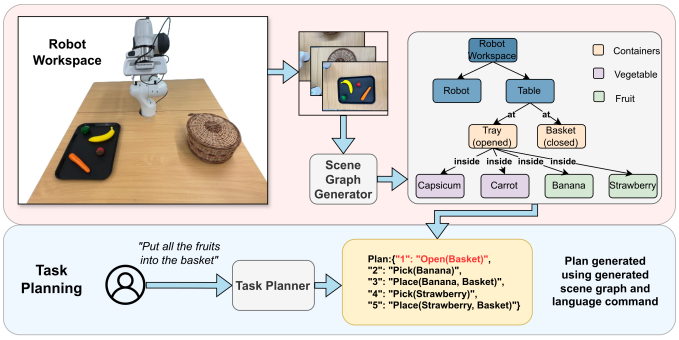

Fig. 1: We address the problem of generating scene graphs in open world settings that capture both metric and semantic information required for robot instruction following tasks. Our factored architecture yields rich accurate scene graphs amenable to rapid update during plan execution.

In response, we develop a factored pipeline where we cascade reasoning as determining object presence, inferring types/spatial extent and finally inducing inter-object relations. Each stage in our pipeline reasons with foundation models using *targeted* (instead of generic) prompts and attended sub-images from the previous stage. We observe that such *foviation* significantly improves the robustness of object detection in the scene graph and subsequently eases relation extraction in relation to approaches that offload the entire scene graph reasoning directly to a large model. Further, we note that while performing sequential manipulation tasks, a robot may need to update its scene model after executing each action. In such a setting, querying a large model on a remote server can introduce significant latency and reduce execution tempo. Hence, we introduce a mechanism for the robot to reason when it encounters a new object (not seen previously), thereby only querying the remote server for such instances and when it can perform reasoning from past memory of objects encountered.

Overall, our results demonstrate generation of robust, accurate scene graphs that are amenable to sequential update. The graphs are more complete in relation to baselines and also reduce construction time when some of the objects have been seen previously during robot operation. This work contributes to the human-robot interaction and long-horizon planning themes in the *ICRA Workshop on Mobile Manipulation and Embodied Intelligence*. Our current experiments are on a robot manipulator but potentially scalable to mobile-manipulation platforms as well.

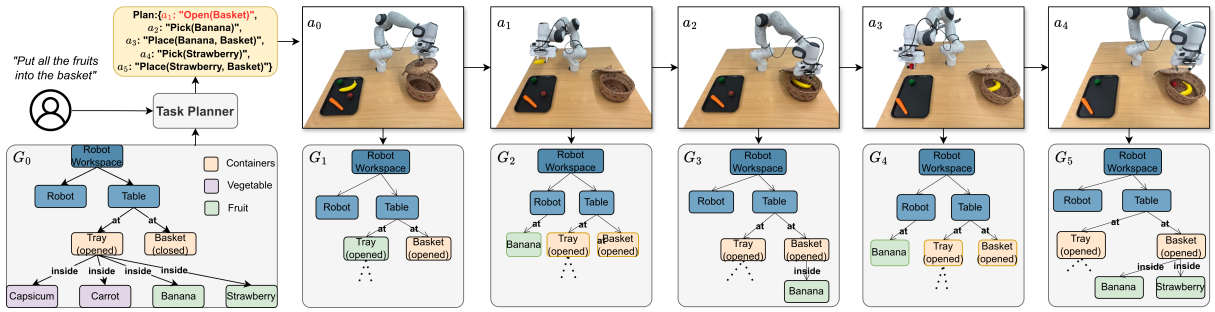

Fig. 2: The figures illustrate the rollout of a plan for the human instruction *"put all the fruits into the basket"*. The task planner synthesizes the plan using the initial scene graph G_0 to generate the sequence of actions a_0 to a_4. The figure also illustrates how this scene representation is amenable to local rebuilding when the scene changes after the robot performs each action. The low-level skill of opening was performed in a semi-autonomous manner.

II. Related Works

Detecting Objects in Scenes. A number of efforts focus on representing the robot's environment as objects with associated metric and semantic properties. They often use supervised object detection models [16], [18], [4], [17], [15] that are trained to identify a narrow predetermined collection of classes. A significant limitation of this method is that, to add or modify the class of identifiable objects, requires the collection and annotation of new data followed by retraining the model. Recent, open vocabulary object detectors [13], [11], [14], [8] can take textual labels as prompts and ground them on the image. However, these models require human inputs such as category names or referring expressions.

Scene Graph Generation Approaches. There have been significant efforts [7], [20], [5] focused on building 3D scene graphs. These scene graphs have a hierarchical structure with different layers representing multiple layers of abstraction from low-level geometry to high-level semantics. These scene graphs also encapsulate the metric information about the entities and are mainly used for navigation related tasks. The OVSG method [2] offers a technique for generating 3D scene graphs to ground free-form text-based queries. However, it depends on graph neural networks to identify relationships between objects, which renders it closed-set with respect to relations. On the other hand, ConceptGraphs [6], creates 3D graphs for room like environments. However, its robustness is limited for workspaces with objects arranged in numerous and less structured settings and the approach does not explicitly consider scene graph update after action execution.

Detecting Scene Elements via VQA. Visual Question Answering (VQA) models [10], [9], [12], [3] learn to associate language instructions with visual observations. Such models can be leveraged for the task of identifying scene objects. However, their key limitation is that they can only generate textual label for the object but not ground them on the image. Since these models are trained to be used as image captioners, they fail to provide granular details of the image which is pivotal for multi-object detection in an image. Recently, models [19], [1], [21] have shown grounding capabilities in addition to generating the textual label for the objects in the image. However, they lack robustness and miss subtle features like text on objects (e.g., medicine labels) required during instruction following.

III. Problem Setup

Consider the robot operating in an environment capable of capturing images from a camera mounted on the arm. The workspace is populated by *a-priori* set of objects $o \in O$. A human instructor provides a natural language instruction $\lambda \in \Lambda$ where Λ is the set of instructions. λ is a combination of objects or object categories and the interactions to be performed with them. Relaxation of this assumption is a part of the future work. The objects in the scene can be contained within (inside), supported by (on top) or be direction-ally oriented (left, right, etc). The robot's overall task is to synthesize the sequence of actions $\pi = [a_0, a_1, ..., a_n]$ in response to the instruction λ and the scene graph G.

In order to support a rich set of instructions from a human, the robot must construct a scene representation that facilitates the understanding of the intended goal from the instruction and facilitates plan synthesis to accomplish the intended goal. Formally, we seek a scene graph that models objects $\mathcal{O}$ and semantic relationships $\mathcal{R}$ between objects arising from rich inter-object interactions such as one object containing or supporting another. The robot must infer a scene graph $\mathcal{G}_t$ at time t from a sequence of RGB-D information $\mathcal{I}_{o:T}$ captured by the robot at certain poses $\mathcal{P}_{o:T}$.

$$\text{SceneGraphGeneration}(\mathcal{I}_{\text{o:T}}) \rightarrow \mathcal{G}_T = (\mathcal{O}, \mathcal{R}) \quad (1)$$

IV. Technical Approach

Directly generating scene graphs from the images $\mathcal{I}_{o:T}$ often leads to poor results (as confirmed by our experiments). Hence, we factor the graph generation as (i) detecting presence and type of objects and (ii) estimating inter-object relations.

$$\text{ObjectDetection}(\mathcal{I}_{\text{o:T}}) \rightarrow \mathcal{O} \quad (2)$$

$$\text{RelationEstimation}(\mathcal{I}_{\text{o:T}} | \mathcal{O}) \rightarrow \mathcal{R} \quad (3)$$

Modeling Objects. Open vocabulary object detection models necessitate prior knowledge about object labels, which limits their effectiveness in unfamiliar settings. To address this limitation, we propose the integration of a Visual Question Answering (VQA) model, which is designated

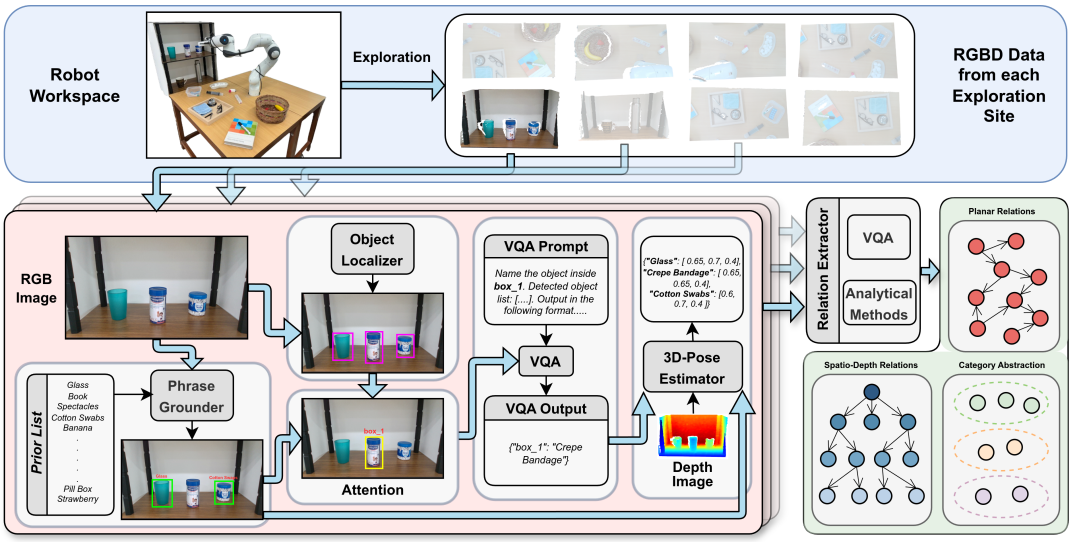

Fig. 3: **Overall Pipeline:** The figure illustrates the generation of a scene graph for a specified robot workspace. Object detection is accomplished through a combination of an object grounding model and a Visual Question Answering (VQA) model. Objects that the grounding model fails to detect are queried to the VQA model through the generation of attention sites. Relations are then extracted using the detected objects as priors to generate multi-hierarchical scene graphs, encompassing category abstraction and spatial-depth and planar relations.

to ascertain object labels. However, it has been observed that VQA models often fall short in identifying all object labels. This shortcoming is attributed to their intrinsic design, which is geared towards offering high-level descriptions or captions of images, rather than detailing granular features. To overcome this shortcoming of VQA models, we developed a mechanism that provides attention to VQA models that can be in the form of unlabelled bounding boxes, masks, cropped images, etc. We detect objects in each image I_t, where $0 < t < T$, using the open vocabulary detectors $\mathcal{H}(\mathcal{I}_t, \mathcal{O}_{0:t-1})$. The objects that $\mathcal{H}(.)$ failed to detect will become the attention sites $\mathcal{A}_t = \mathcal{F}(\mathcal{I}_t) - \mathcal{H}(\mathcal{I}_t, \mathcal{O}_{0:t-1})$ where $\mathcal{F}(\mathcal{I}_t)$ detects all the entities as shown in Fig. 3. The VQA model $\mathcal{V}(\mathcal{A}_t)$ then generates the labels for all the attention sites. The objects detected in the current image $\mathcal{I}_t$ is given by $\mathcal{O}_t = \mathcal{H}(\mathcal{I}_t, \mathcal{O}_{0:t-1}) \cup \mathcal{V}(\mathcal{A}_t)$.

Modeling Relations. Relations between the detected objects $\mathcal{O}_{0:T}$ are found out in a stage-wise manner. We leverage foundation models for extracting spatio-depth relations (e.g. "onTop", "inside", "at") and category abstractions (e.g. "medicinal items", "fruits") and analytical methods for planar relations (e.g. "left", "front"). A VQA model $\mathcal{Q}(\mathcal{I}_{0:T}, \mathcal{O}_{0:T})$ is employed to extract spatio-depth relations $\mathcal{S}$ and category abstractions $\mathcal{C}$ in a hierarchical manner. Planar relations are then calculated analytically $\mathcal{P}(\mathcal{O}_{0:T}|\mathcal{S})$. Relations $\mathcal{R}$ is then given by $\mathcal{R} = \mathcal{S} \cup \mathcal{P} \cup \mathcal{C}$. Figure 4 shows the spatio-depth relations, planar relations and category abstraction for a given robot workspace.

V. Experimental Setup

We conduct experiments on a 7DoF Franka Panda Emika robotic arm with a parallel jaw gripper and an Intel Realsense D435i RGBD camera with eye-in-hand calibration. The robot workspace includes a table and a vertical rack with 2 shelves. We collect a dataset containing 30 different objects commonly found in household environments. These objects were from various categories like *medicinal items* (syringe, pill blister, etc.), *food items* (banana, carrot, etc.), *containers* (basket, tray, etc.), and *personal items* (spectacles, comb, etc.). The objects were also classified into two broad categories - objects that could be identified from their visual features like shape and color (e.g. spectacles, banana) and objects that could only be identified by the textual labels on them (e.g. syrup bottles, hand sanitizer). For the grounding model $\mathcal{H}(.)$, we have used Grounding-DINO[13]; for the object localizer $\mathcal{F}(.)$, we have also used Grounding-DINO; and for Visual Question Answering $\mathcal{V}(.)$, we have utilized OpenAI GPT-4V. However, due to the modular structure of our approach, it can easily be adapted to use other VQA/grounding models.

VI. Results

Quantitative Results. We evaluate the accuracy of our model in detecting objects and their attributes by comparing our pipeline against five baseline methods. The baselines can be segregated into three broad categories: methods that use VQA for labeling and a phrase grounding model for localization, methods employing only VQA for both grounding and labeling, and other state-of-the-art (SOTA) scene graph generators. The first baseline combines a phrase grounder, Grounding DINO, with a VQA model, GPT-4V. The second baseline employs the same framework but utilizes CogVLM for phrase grounding. The third baseline exclusively utilizes CogVLM for both VQA and grounding tasks. The fourth baseline is ConceptGraphs[6] and the fifth baseline is ConceptGraphs-D, a variant of ConceptGraphs which employs an image tagging model(RAM[22]) and a grounding model(Grounding DINO[13]).

Our assessment focuses on two key metrics: robustness and accuracy in object detection. We present our findings

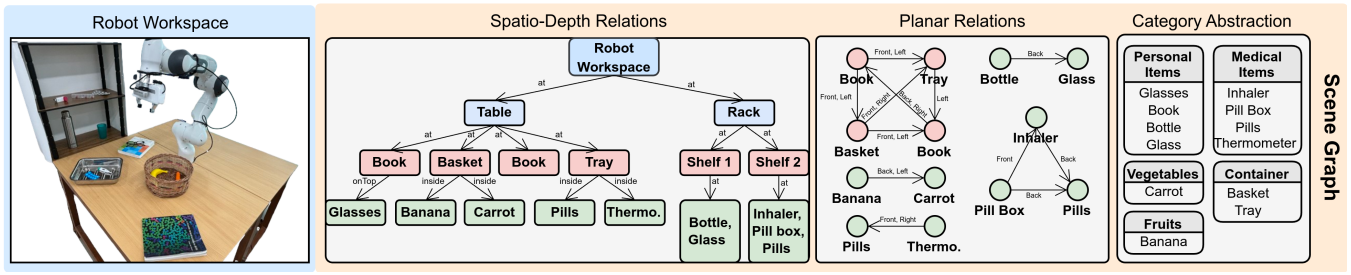

Fig. 4: **Qualitative Result for Scene Graph:** The figure shows the scene graph that was generated by our approach for the given robot workspace. The scene graph has 3 perspectives to it - spatio-depth relations, planar relations, and category-wise abstraction. Each node in the scene graph represents an object in the workspace and has attributes such as color and pose. This scene graph supports the execution of tasks like *"place the book that is to the right of the basket on the rack"*(requires knowledge about the planar relation between the books and the basket and also the 'onTop' relation between book and glasses to generate viable plans), *"give me something to eat"*(requires object category abstraction).

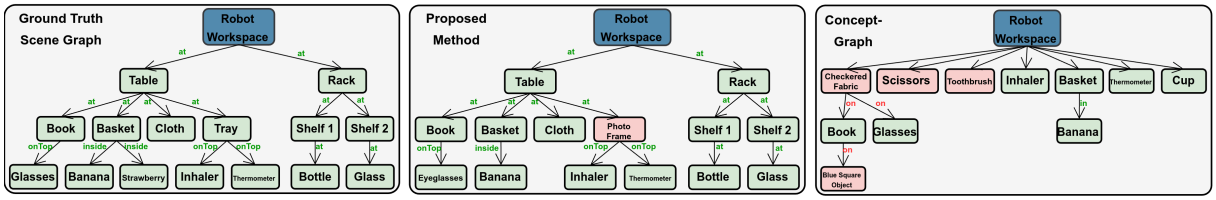

Fig. 5: **Qualitative Comparison of Scene Graphs:** The figure shows the scene graphs generated by both the proposed method and ConceptGraph, alongside the ground truth.

Method	Precision (%)	Recall (%)	F-measure
GDino+GPT-4V	85.0	57.0	0.68
CogVLM+GPT-4V	62.1	93.8	0.75
CogVLM	79.5	77.8	0.79
ConceptGraphs	44.8	73.2	0.54
ConceptGraphs-D	55.8	38.7	0.46
Ours	**94**	**96**	**0.95**

TABLE I: Accuracy of **open-world object detection** during scene graph construction.

across five different indoor settings, each featuring 12-15 randomly sampled objects from our dataset. TABLE I demonstrates that our proposed method achieves the highest F-measure compared to the baseline methods. The performance can be attributed to the method's inherent robustness. In instances where the phrase grounder fails to detect objects, the Visual Question Answering (VQA) model serves as a fail-safe, effectively compensating for these detection lapses. This dual-detection mechanism endows our method with an inherent Optical Character Recognition (OCR) capability, a feature that the baseline methods do not possess. However, in the baselines where both VQA and phrase grounder are present, due to the lack of attention sites, the VQA model hallucinates leading to errors that propagate to the next stage of grounding.

Qualitative Results. Next, we qualitatively evaluate how well our method estimates spatio-depth relations ("onTop", "inside", "at"), planar relations ("left", "right", "front", "back"), and categorical abstractions. Fig. 4 shows the multi-hierarchical scene graph that was generated for an indoor setting, with 14 objects randomly sampled from the dataset. Fig. 5 shows the qualitative comparison between the scene graphs generated by the proposed method and ConceptGraph. Specifically, we focused on spatio-depth relations for this comparison due to ConceptGraph's limitation in generating planar relations and categorical abstractions. Our results indicate that the proposed method demonstrates robustness in object detection, consequently leading to more accurate relation extraction. However, the proposed method made one incorrect detection and missed one object. In contrast, ConceptGraph missed three objects, mislabeled two objects, and also produced some hallucinations.

VII. Conclusions

We introduce a novel approach for generating multi-hierarchical scene graphs for open-world settings which facilitates the execution of sequential and long-horizon plans. It employs a factored approach to scene graph generation, first detecting objects and then establishing relations between them. Experimental evaluations showcase the effectiveness of our approach in generating rich and accurate scene graphs without prior knowledge of objects. The graphs are more robust in comparison to baseline methods and can be updated rapidly during a multi-step manipulation task execution. Ongoing work focuses on more extensive evaluation. Future work will extend the approach to mobile manipulators.

ACKNOWLEDGEMENTS

We are grateful to Mr. Himanshu Gaurav Singh for assisting in setting up the LLM/LVM infrastructure (while being student at IIT Delhi) and feedback on this work (after graduating). We thank Mr. Mohd. Nadir, and Dr. Piyush Chanana for conceptualizing and realizing the Franka Emika Panda manipulation test bed used for real experiments. We acknowledge Mr. Shailendra Negi and Ms. Sunita Negi for administrative support. We thank Namasivayam K and the IIT Delhi HPC team for assisting with setup and maintenance of high-performance computing infrastructure used for this work. Rohan Paul and P. V. M. Rao acknowledge research funding support from IIT Delhi IRD-Unit, NCAHT-ICMR and DIA-COE.

REFERENCES

[1] J. Bai, S. Bai, S. Yang, S. Wang, S. Tan, P. Wang, J. Lin, C. Zhou, and J. Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. 2023.

[2] H. Chang, K. Boyalakuntla, S. Lu, S. Cai, E. Jing, S. Keskar, S. Geng, A. Abbas, L. Zhou, K. Bekris, et al. Context-aware entity grounding with open-vocabulary 3d scene graphs. *arXiv preprint arXiv:2309.15940*, 2023.

[3] W. Dai, J. Li, D. Li, A. M. H. Tiong, J. Zhao, W. Wang, B. Li, P. N. Fung, and S. Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36, 2024.

[4] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2147–2154, 2014.

[5] E. Greve, M. Büchner, N. Vödisch, W. Burgard, and A. Valada. Collaborative dynamic 3d scene graphs for automated driving. *arXiv preprint arXiv:2309.06635*, 2023.

[6] Q. Gu, A. Kuwajerwala, S. Morin, K. M. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. *arXiv preprint arXiv:2309.16650*, 2023.

[7] N. Hughes, Y. Chang, and L. Carlone. Hydra: A real-time spatial perception system for 3d scene graph construction and optimization. *arXiv preprint arXiv:2201.13360*, 2022.

[8] A. Kamath, M. Singh, Y. LeCun, G. Synnaeve, I. Misra, and N. Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1780–1790, 2021.

[9] J. Li, D. Li, S. Savarese, and S. Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.

[10] J. Li, D. Li, C. Xiong, and S. Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[11] L. H. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J.-N. Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022.

[12] H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

[13] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[14] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen, et al. Simple open-vocabulary object detection with vision transformers. arxiv 2022. *arXiv preprint arXiv:2205.06230*, 2, 2022.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[16] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[17] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[18] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. *Advances in neural information processing systems*, 26, 2013.

[19] W. Wang, Q. Lv, W. Yu, W. Hong, J. Qi, Y. Wang, J. Ji, Z. Yang, L. Zhao, X. Song, et al. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*, 2023.

[20] S.-C. Wu, J. Wald, K. Tateno, N. Navab, and F. Tombari. Scenegraphfusion: Incremental 3d scene graph prediction from rgb-d sequences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7515–7525, 2021.

[21] H. You, H. Zhang, Z. Gan, X. Du, B. Zhang, Z. Wang, L. Cao, S.-F. Chang, and Y. Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*, 2023.

[22] Y. Zhang, X. Huang, J. Ma, Z. Li, Z. Luo, Y. Xie, Y. Qin, T. Luo, Y. Li, S. Liu, et al. Recognize anything: A strong image tagging model. *arXiv preprint arXiv:2306.03514*, 2023.

APPENDIX

A. Detecting Objectness

We employ a factored approach for detecting novel objects in a scene, where the presence of objects in an image is initially identified using foundational models termed as objectness. The subsequent step involves determining the labels for these objects. To detect objectness in an image, we utilize a generic prompting technique, employing "objects" as the prompt for the Grounding-DINO model, which localizes all possible entities in the given image. However, detecting objectness is not confined solely to this prompting strategy; alternative methods such as leveraging class-agnostic segmentation models like SAM can also be utilized.

Fig. 6: *"objects"* prompt for Grounding DINO for detecting objectness

B. Prompts for Hierarchical Relation Estimation

A Visual Question Answering (VQA) model is employed to extract the hierarchical relationships between the detected objects. We have explored various prompting strategies to induce this hierarchy using foundation models. The VQA used for the experiments is GPT-4V, which has the capability to accept multiple images as inputs in a single query. The following shows the prompt that we used:

```
Objects in the image are: [<obj 1>, <obj 2>, ...].
Find the relations (inside, onTop) between all
the objects mentioned using the images.
Not all the objects will be present in each image.
The objects can be either on the table (light brown color)
or on a rack. For each object, also tell if the
object is at the table or at the rack.
```

We have not constrained the VQA model to output in a structured format. Experiments have shown that constraining the output prevents the model from engaging in inherent Chain-of-Thought reasoning, resulting in

hallucinations. To structure the output, we employ an LLM parser, which induces hierarchy and categorizes object types.

C. Comparison of Scene Graph Generation Methods

The Scene Graph encompasses more than one aspect, including the name and pose of objects, as well as relations such as spatio-depth, planar, and category abstraction. However, not all methods can handle all of these aspects. TABLE II illustrates the capabilities of each baseline method.

Method	Objects		Relations		
	Detection	Pose	Spatio-depth	Planar	Abstraction
GPT-4V	✓	✗	✓	✗	✓
CogVLM	✓	✓	✗	✗	✗
GPT-4V+GDINO	✓	✓	✓	✓	✓
GPT-4V+CogVLM	✓	✓	✓	✓	✓
ConceptGraph	✓	✓	✓	✗	✗
Proposed	✓	✓	✓	✓	✓

TABLE II: Comparison of various method for generating scene graphs

D. Time Efficiency

Since querying the VQA model is computationally expensive, our method caches all previous detections so that in the next iteration, querying the VQA model can be reduced. However, since the baselines always rely on the VQA models for object labeling, there is no added benefit of caching. Fig. 7 shows the number of VQA calls made by the proposed method, GPT-4V+GDINO (Baseline 1), and GPT-4V+CogVLM (Baseline 2) over two iterations of scene graph generation.

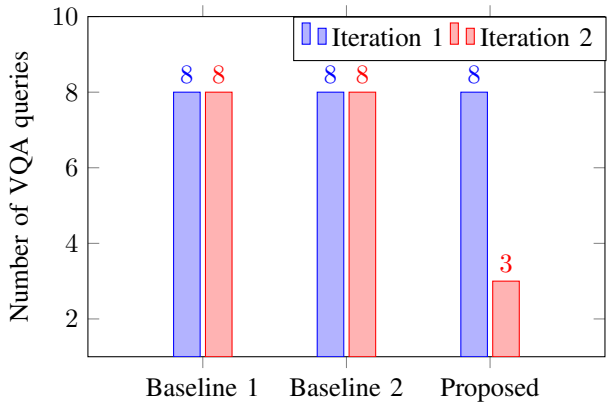

Fig. 7: Optimality in Object Detection