A Survey on 3D Scene Graphs: Definition, Generation and Application

Jaewon Bae*, Dongmin Shin*, Kangbeen Ko, Juchan Lee, and Ue-Hwan Kim

GIST, 123 Cheomdangwagi-ro, Buk-gu, Gwangju 61005, Republic of Korea
{jaewonbae, newdm2000}@gm.gist.ac.kr,
{eyeoftyphoon, jclee0109, uehwan}@gist.ac.kr,
https://uehwan.github.io

Abstract. With the advancement of intelligent agents, 3D scene understanding has become one of key tasks of computer vision. 3D scene understanding is challenging to represent effectively because objects form various relationships and constantly interact with each other. A scene graph is a powerful tool to concisely represent the properties and relationships of objects in a scene—enabling various multi-modal tasks. Therefore, research on 3D scene graph (3DSG) is attracting increasing attention. However, 3DSG research is in its early stage—requiring a systematically organized survey. In this paper, we survey the latest advancement of 3DSG. In addition, we clarify 3DSG concepts that are currently defined in various ways, provide real-world applicability and present future research directions.

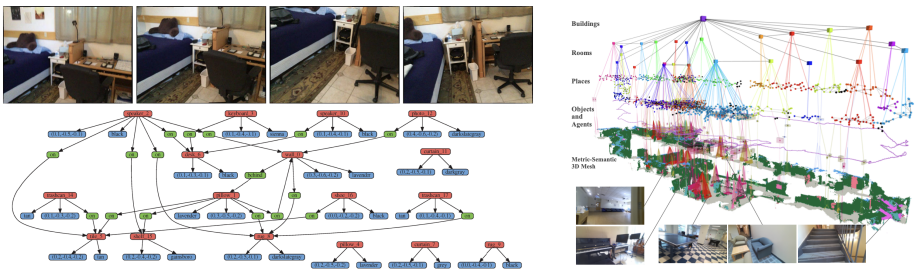
Keywords: 3D Scene Graph, 3D Scene Understanding, Visual Relationship Detection, Robot Vision

1 Introduction

The research on visual scene understanding aims to make machines understand scenes to that of human-level; humans can recognize the gist of a scene in a very short time and can infer deeper information from the scene [1]. Visual scene understanding research has shifted its focus from simple object-centric recognition tasks such as image classification [2], object detection [3], and semantic segmentation [4] to high-level tasks that infer relationships and interactions between objects. In addition, researchers have gradually expanded the target domain from 2D visual data to 3D visual data for real world applications. In-depth recognition of 3D environments is indispensable for intelligent agents such as robots and autonomous driving machines to perform complex tasks or provide useful services to humans.

3D environments involves numerous objects, and each object forms an organic relationship with each other. Thus, effective representations for 3D environments play a crucial role for intelligent agents to gather and store information from the

* The two authors contributed equally to this paper.



(a) (b)

Fig. 1: Examples of 3D scene graph (3DSG). (a) Flat 3DSG example [5]. (b) Hierarchical 3DSG example [6].

environments. Among various attempts to realize an effective representation for 3D environments, 3D scene graphs (3DSGs) have emerged recently. 3DSG is a knowledge graph and framework that detects objects in the 3D scene and predicts the relationship between them—able to effectively represent the 3D scenes.

Scene graph (SG) was widely used in computer graphics as a hierarchical representation for complex 3D scenes [7, 8]. Taking this into consideration, Johnson, et al. [9] introduced SG for the first time in the field of computer vision in 2015. In order to understand the semantics of images, the SG proposed represents visually-grounded graphs by extracting objects, attributes, and relationships from images, respectively. Then, they applied SG to the image retrieval task to prove its usefulness. Since then, research on automatic scene graph generation (SGG)[1] from an image or video has drawn much attention: SGG-related research on 2D image input [10–15], and SGG-related research on 2D video input [16–20]. 3DSG is an extension of 2DSG technology into three dimensions, which first appeared in 2018 [21].

3DSG has gained rapid popularity in a short period of time, but an in-depth survey of 3DSG is not yet available. Although some review papers dealing with overall SG included 3DSG, analysis of 2DSG still dominates. For example, 3DSG is in some cases only regarded as an application example of SG [22] or a small branch according to the modalities of SG's input (i.e., Image, Video and 3D mesh) [23]. The proportion of 3DSG is considerably small compared to its significance and the latest studies are lacking. We intensively survey and systematize 3D scene graph generation (3DSGG) methodologies, datasets, and applications in this article—providing researchers with insight into the latest 3DSG research.

The rest of this paper is organized as follows. Section 2 summarizes the concept of general SG and classifies 3DSG into two major groups (i.e., flat, hierarchical). Section 3 introduces 3DSG benchmark datasets and evaluation metrics. Section 4 examines the applicability of 3DSG, and demonstrates the

[1] Note that SG generation (SGG) or SG prediction or SG construction are interchangeable.

Table 1: A comparison of SGG

	Input	Attribute	Output	Generator
2DSG	2D Image	Class	Flat 2DSG	[10–15]
3DSG	RGB-D Image Point Cloud 3D Mesh Voxel	Class 3D Position Volume Height	Flat 3DSG	[21, 5, 24, 25]
			Hierarchical 3DSG	[26–28, 6]

effectiveness of 3DSG in a 3D environment. Section 5 discusses potential future research directions. Conclusion follows in Section 6 with important remarks.

2 3D Scene Graph Generation

In this section, we first summarize the concept of SGG and describe the difference between 2DSGs and 3DSGs. Furthermore, we focus on examining the 3DSGG methods and classify them according to the graph types. We also introduce advanced techniques proposed to improve the performance of 3DSGG.

2.1 Overview

SGG aims to fully understand a scene by analyzing an input image or video to generate a structured representation of the scene as a graph format—which is very challenging. The basic expression unit of SGs is the <*subject-relation-object*> triplet. Each triplet gets reconnected to others to form the entire SG for a scene. Thus, SGG is a bottom-up process. SGG process includes 1) object detection, which recognizes objects in the given scene, 2) relation extraction, which captures the relations between objects, and 3) graph generation, which forms a graph by connecting the triplets. Through this SGG process, we can finally obtain 2DSGs or 3DSGs. This can be divided according to whether the input data is 2D or 3D (Table 1).

Conventional 2DSGs lack sufficient representation of the complex interaction of objects in the 3D space (no 3D position and relation information). To overcome this limitation, 3DSG has emerged. 3DSG can represent various interactions in the real world with simple and effective expression of the shape, position, and properties of objects in 3D spaces.

In other words, 3DSGG is a technique that receives scenes containing 3D information as inputs and outputs 3DSG. There are two main categories of 3DSG representations: 1) flat 3DSG and 2) hierarchical 3DSG. The flat 3DSG represents the object and relation detected in a 3D scene as a simple graph similar to 2DSG; the hierarchical 3DSG represents the structures of 3D scenes by layers (e.g., rooms, floors and buildings).

2.2 Flat 3DSG

Gay et al. [21] first introduced the term *3D scene graph* in computer vision along with the formulation of the task for prediction of 3DSG in multi-view. Further, they proposed to generate 3DSGs by estimating 3D location and object occupancy from 2D bounding boxes. However, this work did not clearly describe the essential difference between 3DSG and 2DSG.

Next, Kim et al. [5] pointed out the limitations of conventional 2DSG methods, such as narrow applicability, unintuitive usability, and low scalability, and defined a 3DSG for representing 3D scenes with 3D information such as position. Further, the paper proposed a 3DSG construction framework and proved the applicability of 3DSG by demonstrating the effective 3D scene understanding of intelligent agents using 3DSG—signifying the contribution of the paper beyond simple accuracy but applicability. However, there is a limitation that various levels of abstraction couldn't be provided due to the limited types of nodes and relationships.

On the other hand, researchers have conducted generating 3DSG from 3D input data rather than image-level input forms. Wald et al. [24] presented a method of generating 3DSG with a 3D point cloud as input, and proved that 3DSG can resolve the 2D and 3D domain gap by serving as a shared domain between 2D and 3D. In addition, they opened the 3DSSG 3D Semantic Scene Graphs, a large-scale real-world dataset, for vitalization of research in the relevant field. Moreover, Zhang et al. [25] presented a methodology for predicting SG directly from 3D point cloud by encoding the point cloud into a more distinguishable latent space as prior knowledge.

2.3 Hierarchical 3DSG

Previous papers presented flat 3DSGs, which in some cases did not sufficiently express the hierarchy between objects or spaces. To resolve this issue, Armeni et al. [26] presented a methodology for generating a hierarchical 3DSG that semantically classifies objects using a panoramic view and 3D mesh constructed from input RGB-D images. Further, they proposed a semi-automatic framework that improves the performance of the semantic hierarchy recognition by using framing of query images for better performance of 2D detector and applying multi-view consistency from different camera locations.

Rosinol et al. [27] proposed a dynamic 3DSG that additionally represents dynamic entities and relations to reduce the gap in cognitive ability for scenes between humans and robots. In addition, they presented the Kimera framework, which is an end-to-end learning method for generating a complete dynamic 3DSG from visual inertial data.

2.4 Advanced Techniques for 3DSG

Incremental 3DSG. Intelligent agents would incrementally explore the environment and the scope of the environment of interest would expand. Thus, 3DSG

should be able to account for both unseen physical attributes and novel semantic entities. Accordingly, Wu et al. [28] proposed a method to incrementally generate 3DSG from RGB-D images. For this, they designed an attention-based graph processing mechanism that combines incrementally incoming 3DSG and recognizing relations not detected in previous steps. Similarly, Hughes [6] presented a real-time 3DSG generation method using top-down loop closure detection with a hierarchical descriptor that captures statistics across SG layers for optimizing the entire 3DSG

Li et al. [29] proposed a method of generating the local SG according to the action of the agent and adding it to the global SG incrementally. In addition, they proposed a navigation framework that feeds the concatenated features generated from the RGB frame, local scene graph, and action type at the current time point t to generate the pertinent action at the next time point $t + 1$.

Miscellaneous. Zhang et al. [30] fused *support information* for robotic manipulation into 3DSG in the form of volumetric representation by adding detailed support relation using simple contact relation graphs.

Zhang et al. [31] enhanced the 3DSG generation performance by sequentially adding reasoning and inferencing steps. Talak et al. [32] presented H-Tree, a tree structured architecture that can replace the conventional method of message passing—focusing on the representation of the input graph. As a result of applying the proposed architecture to the aggregation function of the 3DSG generation models based on various GNN methodologies, the performance significantly improved.

3 Datasets and Evaluation Metrics

Datasets with relevant annotations are essential for learning SGG. A number of datasets providing 2D images [33, 10, 34, 9, 35, 36] or videos [37, 16, 38, 39] for SGG exist. However, there are few datasets for learning 3DSG generation. Kim et al. [5], one of the earliest 3DSG studies, tested the performance of 3DSG generation using ScanNet [40] as an alternative; ScanNet is an RGB-D video dataset without relationship annotation. In this section, we tabularize 3DSG datasets in various fields according to target task types.

3.1 3DSG Datasets

Indoor Scene Understanding Most of the datasets collected from indoor scenes focus on spatial relationships. They aim to construct the overall SG focusing on the positional relationship between objects in a static situation. The types of input data include point clouds or RGB-D sequences. 3DSSG [24] based on 3RScan [47] provides 1,482 RGB-D scans from 478 real-world environments. 3DSSG annotatated SG labels to 3RScan and presented evaluation metrics for 3DSG. The spatial commonsense graph dataset [41] supplies spatial commonsense graphs for partial reconstruction of ScanNet point clouds [40]. Armeni et al. based on the Gibson environment [48], [26] presented the hierarchical 3DSG generation rules and benchmark datasets.

Table 2: A comparison of existing datasets for scene graph generation

	Dataset	Scale	Obj.		Rel.	
			#	Cls.	#	Cls.
Indoor Scene	3DSSG[24]	478 scenes, 1482 scans	48k	534	544k	40
	Giuliari et al. [41]	1201 scenes, 25k RGB-D frames	-	40	-	3
	Armeni et al. [26]	35 Buildings, 727rooms	3k	28	-	4
Action Recognition	Road Scene Graph [42]	505 Scenes	-	23	-	16
	Bimanual Actions [43]	540 recordings	-	16	-	16
	4D-OR [44]	6734 scenes	-	12	-	14
Object-Centric	Rel3D [45]	9990 3D Scenes	-	67	9990	30
	PTR [46]	52k RGB-D images	-	5	-	7

Action Recognition Action recognition datasets focus on estimating temporally-varying relations based on the movement of humans or objects. 3DSG generation based on action recognition datasets aims to generate a SG that reflects the variation of relations or actions according to the passage of time. Thus, these datasets provide videos. Road Scene Graph [42] provides annotated SGs consisting of objects such as cars, motorcycles, and pedestrians in addition to camera images and lidar sensor data. Bimanual actions dataset [43] focuses on the relation between humans and objects while a person performs a specific bimanual task. 3-stage pre-processing of RGB-D video resulted in SG annotations. 4D-OR [44] concentrates on the relations of Human-Object and Object-Object in operation-rooms (OR) in knee surgeries situations. The RGB-D Kinetic sensor was configured as Multi View, and the fused 3D point cloud was extracted.

Object-centric Relationship Detection Due to the high cost of collecting real-world datasets, researchers actively utilize synthetic datasets collected from simulation environments as well. Simulation allows the control of the types of objects, the arrangement of objects, and the size of the resulting dataset. Rel3D Dataset [45] collected RGB images after arranging two objects according to a specific relation in a graphic engine. PTR Dataset [46] aims to perform relation detection between objects and object parts and the Visual Question Answering (VQA) task. For this purpose, objects are arranged in a physical engine, and the relationship between objects, components of objects, and scenes are extracted along with RGB-D images.

Limitation There are still very few datasets for learning 3DSG generation, and some of them are too task-specific to be universally used. In addition, the annotation of the relationship is simple and not sufficiently detailed, so a large-scale richly-annotated 3DSG dataset is necessary.

3.2 Evaluation Metrics

Recall@K Currently, *Recall*@K ($R@K$) is generally used as an evaluation metric. $R@K$ is the ratio of correct predictions out of K most-confident predictions. On the other hand, Tang et al. [49] discussed the usage of mAP, another popular metric along with $R@K$, as an evaluation metric; providing negative assessment because mAP exhaustively evaluates all the relationship in an image. Next, Gkanatsios et al. [50] proposed $Recall_k@x$ ($R_k@x$). In $R_k@x$, k and x represnet the number of predictions for a pair of objects and the most confident x predictions similar to K in $R@K$, respectively. For $k = 1$, it is called *graph* constraints, and otherwise *no graph constraints*.

Mean Recall@K $R@K$ entails a problem with long tail-biased datasets. To address this, Tang et al. [49] proposed $Mean\,Recall@K$ ($mR@K$). $mR@K$ is a method of calculating recalls according to each relation classification and averaging the recalls. For the choice of K of $R@K$ and $mR@K$, $R@20$, $R@50$, and $R@100$ are commonly used; $R@5$ or $R@10$ is also frequently used. However, clear criteria for choosing K have not yet been examined.

Evaluation Task The performance of 3DSG is occasionally evaluated from several intrinsic subtasks [20, 25, 51, 24]. In general, three subtasks are prevailing

- PredCls: a task of predicting the relation between a subject and an object given ground-truth bounding boxes and class labels.
- SGCls: a task of predicting class label and relation given ground-truth bounding boxes.
- SGGen: a task of predicting the object pairs, class labels, and relations for the objects detected by a detector.

zhang et al. [25] evaluated the performance using PredCls and SGCls tasks with $R@K$ and $mR@K$ metrics. Additionally, performance can be further evaluated using with or without constraint conditions [24, 52]. This condition corresponds to the case where $k = 1$ in $Recall_k@x$ ($R_k@x$) and the case where $k > 1$, respectively.

Evaluating the performance of 3DSG generation by intrinsic subtasks has the advantages of computational efficiency and easy understanding of the system. However, the correlation as to whether it will actually aid real world tasks is not clear. Accordingly, there is also a method of evaluating performance with an extrinsic task. Kim et al. [5] evaluated the performance by human judges.

4 Applications of 3D Scene Graph

4.1 Applications in Robotics

Robot Task Planning Robot task planning generates the sequence of atomic actions for a robot to perform a given task. Kim et al. [5] illustrated how 3DSG

can aid task planning with an algorithm that transforms a 3DSG into the problem description required in task planning (i.e., the fast-forward planner). Agia et al. [53] presented the Taskography Benchmark for real-time task planning from 3DSG. Jiao et al. [54] proposed expressing 3DSG as a contact graph for efficient sequential task planning.

Robot Navigation Robot navigation is a task that aims for robots to recognize the 3D environment and navigate successfully to the target point. Ravihandran et al. [55] proposed a graph neural network architecture for learning navigation policies using 3DSG. In addition, they proved that they could enhance the memory usage efficiency of storing the 3DSG in agent-centric and performing the navigation task compared to rendering the entire environment.

Embodied Question Answering Embodied Question Answering (EmbodiedQA) is the task of answering questions regarding 3D environments in natural language. Kim et al. [5] utilized 3DSG to cope with four types of Embodied QA tasks: Object Counting, Counting with attributes, Counting with relations, Multimodal VQA.

4.2 Applications in Computer Vision

Scene Retrieval Scene retrieval is the task of finding a 2D or 3D scene of interest. Wald et al. [24] proposed a cross domain 2D-3D scene retrieval framework where a pre-formed 3DSG aids the retrieval task in 3 domains: 2D, 3D, and natural language.

Scene Generation Scene Generation is the task that creates a scene based on a given input. Dhamo et al. [56] proposed an architecture that creates a 3D scene from SG, and evaluated the generated 3D scene through 3DSSG datasets [24]. Savkin et al. [57] proposed an unsupervised learning-based learning method that can create a traffic 2D synthetic image scene from a 3DSG.

5 Future Research Direction

3DSGG in Dynamic Environment 3DSGG in a 3D environment with dynamic objects still remains a big challenge. In the case of dynamic objects, the accuracy of relation extraction is considerably low; it is difficult to represent a 3D scene as a single deterministic 3DSG with dynamic objects because of the various interpretation possibilities (e.g., *eat-drink-hold*) and time-varying characteristics. Therefore, a 3DSGG method that can reflect real-time movement or environmental changes should be tackled.

Benchmark Datasets Currently, evaluation metrics and benchmarks for 3DSG are insufficient. Although the number of generators and models for 3DSG is increasing every year, there are few datasets in comparison. Each 3DSG generator employs different metrics, making fair comparison difficult. Therefore, researchers should develop a large scale 3DSG benchmark dataset and evaluation metrics. Moreover, the benchmark dataset must include dynamic objects along with a wide range of scenes that can reflect the real world environments.

Lightweight 3DSG Generator Intelligent agents would mainly utilize 3DSGs for various purposes. To achieve various goals of intelligent agents, agents in general should be able to generate a 3DSG in real time processing the input data collected through the agent's sensors. However, intelligent agents such as service robots are ordinarily equipped with limited computational resources. Therefore, a lightweight 3DSG generator that requires a small amount of computation resources and generates 3DSG in real-time demands further research.

Spatial Ambiguity of 3DSG When generating the 3DSG in a 3D environment, the spatial relation is relative according to the location of the agent or the observer. For example, for two objects positioned side by side, the relation between the two objects is 'next to' when viewed from the front while the relation is 'in front of' when viewed from the side. This spatial ambiguity problem does not occur in 2D images where the camera pose is fixed, but it does frequently occur when defining 3DSGs. Such spatial ambiguity would cause malfunctions of intelligent agents when performing tasks. Therefore, an accurate representation of spatial and positional relationship would resolve the ambiguity.

6 Conclusion

In this work, we conducted a systematic literature survey on 3DSG research. We first introduced various formulations and definitions of 3DSG: flat and hierarchical representations. Then, we categorized conventional datasets for 3DSG generation and evaluation metrics. Furthermore, we discussed the applications of 3DSG such as robot task planning, robot navigation and scene generation. 3DSG enhances the performance of these tasks due to its concise and rich representation of 3D scenes. Last, we provided crucial insights toward future research direction: despite active research conducted for 3DSG, the accuracy of relation detection is too low to guarantee real-world deployment and the overall performance highly depends on the performance of object detection. We expect our work would provide well-organized information on 3DSG and aid future research on 3DSG.

Acknowledgement This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-00907, Development of AI Bots Collaboration Platform and Self-organizing AI)

References

1. G. J. Zelinsky, "Understanding scene understanding," 2013.
2. W. Wang, Y. Yang, X. Wang, W. Wang, and J. Li, "Development of convolutional neural network and its application in image classification: a survey," *Optical Engineering*, vol. 58, no. 4, p. 040901, 2019.
3. S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, p. 103514, 2022.
4. X. Liu, Z. Deng, and Y. Yang, "Recent progress in semantic image segmentation," *Artificial Intelligence Review*, vol. 52, no. 2, pp. 1089–1106, 2019.
5. U.-H. Kim, J.-M. Park, T.-J. Song, and J.-H. Kim, "3-d scene graph: A sparse and semantic representation of physical environments for intelligent agents," *IEEE transactions on cybernetics*, vol. 50, no. 12, pp. 4921–4933, 2019. `https://github.com/Uehwan/3-D-Scene-Graph`.
6. N. Hughes, Y. Chang, and L. Carlone, "Hydra: a real-time spatial perception system for 3d scene graph construction and optimization," 2022.
7. M. Fisher, M. Savva, and P. Hanrahan, "Characterizing structural relationships in scenes using graph kernels," in *SIGGRAPH*, pp. 1–12, 2011.
8. R. F. Tobler, "Separating semantics from rendering: a scene graph based architecture for graphics applications," *The Visual Computer*, vol. 27, no. 6, pp. 687–695, 2011.
9. J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei, "Image retrieval using scene graphs," in *CVPR*, pp. 3668–3678, 2015.
10. C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei, "Visual relationship detection with language priors," in *ECCV*, pp. 852–869, Springer, 2016.
11. D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene graph generation by iterative message passing," in *CVPR*, pp. 5410–5419, 2017.
12. Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang, "Scene graph generation from objects, phrases and region captions," in *ICCV*, pp. 1261–1270, 2017.
13. J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh, "Graph r-cnn for scene graph generation," in *ECCV*, pp. 670–685, 2018.
14. Y. Li, W. Ouyang, B. Zhou, J. Shi, C. Zhang, and X. Wang, "Factorizable net: an efficient subgraph-based framework for scene graph generation," in *ECCV*, pp. 335–351, 2018.
15. K. Tang, Y. Niu, J. Huang, J. Shi, and H. Zhang, "Unbiased scene graph generation from biased training," in *CVPR*, pp. 3716–3725, 2020.
16. X. Shang, T. Ren, J. Guo, H. Zhang, and T.-S. Chua, "Video visual relation detection," in *ACM Multimedia*, October 2017.
17. Y.-H. H. Tsai, S. Divvala, L.-P. Morency, R. Salakhutdinov, and A. Farhadi, "Video relationship reasoning using gated spatio-temporal energy graph," in *CVPR*, pp. 10424–10433, 2019.
18. Y. Teng, L. Wang, Z. Li, and G. Wu, "Target adaptive context aggregation for video scene graph generation," in *CVPR*, pp. 13688–13697, 2021.
19. Y. Cong, W. Liao, H. Ackermann, B. Rosenhahn, and M. Y. Yang, "Spatial-temporal transformer for dynamic scene graph generation," in *CVPR*, pp. 16372–16382, 2021.
20. Y. Li, X. Yang, and C. Xu, "Dynamic scene graph generation via anticipatory pre-training," in *CVPR*, pp. 13874–13883, 2022.

21. P. Gay, J. Stuart, and A. Del Bue, "Visual graphs from motion (vgfm): Scene understanding with object geometry reasoning," in *ACCV*, pp. 330–346, Springer, 2018. `https://github.com/paulgay/VGfM`.
22. X. Chang, P. Ren, P. Xu, Z. Li, X. Chen, and A. G. Hauptmann, "A comprehensive survey of scene graphs: Generation and application," *TPAMI*, 2021.
23. G. Zhu, L. Zhang, Y. Jiang, Y. Dang, H. Hou, P. Shen, M. Feng, X. Zhao, Q. Miao, S. A. A. Shah, *et al.*, "Scene graph generation: A comprehensive survey," *arXiv preprint arXiv:2201.00443*, 2022.
24. J. Wald, H. Dhamo, N. Navab, and F. Tombari, "Learning 3d semantic scene graphs from 3d indoor reconstructions," in *CVPR*, pp. 3961–3970, 2020. `https://3dssg.github.io/#download`.
25. S. Zhang, A. Hao, H. Qin, *et al.*, "Knowledge-inspired 3d scene graph prediction in point cloud," *NeurIPS*, vol. 34, pp. 18620–18632, 2021.
26. I. Armeni, Z.-Y. He, J. Gwak, A. R. Zamir, M. Fischer, J. Malik, and S. Savarese, "3d scene graph: A structure for unified semantics, 3d space, and camera," in *CVPR*, pp. 5664–5673, 2019. `https://github.com/StanfordVL/3DSceneGraph`.
27. A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, and L. Carlone, "Kimera: From slam to spatial perception with 3d dynamic scene graphs," *International Journal of Robotics Research*, vol. 40, no. 12-14, pp. 1510–1546, 2021. `https://github.com/MIT-SPARK/Kimera`.
28. S.-C. Wu, J. Wald, K. Tateno, N. Navab, and F. Tombari, "Scenegraphfusion: Incremental 3d scene graph prediction from rgb-d sequences," in *CVPR*, pp. 7515–7525, 2021.
29. X. Li, D. Guo, H. Liu, and F. Sun, "Embodied semantic scene graph generation," in *CoRL*, pp. 1585–1594, PMLR, 2022.
30. P. Zhang, X. Ge, and J. Renz, "Support relation analysis for objects in multiple view rgb-d images," in *IJCAI*, pp. 41–61, Springer, 2019.
31. C. Zhang, J. Yu, Y. Song, and W. Cai, "Exploiting edge-oriented reasoning for 3d point-based scene graph analysis," in *CVPR*, pp. 9705–9715, 2021.
32. R. Talak, S. Hu, L. Peng, and L. Carlone, "Neural trees for learning on graphs," *NeurIPS*, vol. 34, pp. 26395–26408, 2021.
33. R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *IJCV*, vol. 123, no. 1, pp. 32–73, 2017.
34. A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov, *et al.*, "The open images dataset v4," *IJCV*, vol. 128, no. 7, pp. 1956–1981, 2020.
35. Y. Liang, Y. Bai, W. Zhang, X. Qian, L. Zhu, and T. Mei, "Vrr-vg: Refocusing visually-relevant relationships," in *CVPR*, pp. 10403–10412, 2019.
36. J. Yang, Y. Z. Ang, Z. Guo, K. Zhou, W. Zhang, and Z. Liu, "Panoptic scene graph generation," *arXiv preprint arXiv:2207.11247*, 2022.
37. J. Ji, R. Krishna, L. Fei-Fei, and J. C. Niebles, "Action genome: Actions as compositions of spatio-temporal scene graphs," in *CVPR*, pp. 10236–10247, 2020.
38. X. Shang, D. Di, J. Xiao, Y. Cao, X. Yang, and T.-S. Chua, "Annotating objects and relations in user-generated videos," in *ICMR*, pp. 279–287, 2019.
39. T. Zhuo, Z. Cheng, P. Zhang, Y. Wong, and M. Kankanhalli, "Explainable video action reasoning via prior knowledge and state transitions," in *ACM Multimedia*, pp. 521–529, 2019.

40. A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *CVPR*, pp. 5828–5839, 2017. http://www.scan-net.org/.
41. F. Giuliari, G. Skenderi, M. Cristani, Y. Wang, and A. Del Bue, "Spatial commonsense graph for object localisation in partial scenes," in *CVPR*, pp. 19518–19527, 2022. https://fgiuliari.github.io/projects/SpatialCommonsenseGraph/.
42. Y. Tian, A. Carballo, R. Li, and K. Takeda, "Road scene graph: A semantic graph-based scene representation dataset for intelligent vehicles," *arXiv preprint arXiv:2011.13588*, 2020. https://github.com/tianyafu/road-status-graph-dataset.
43. C. R. Dreher, M. Wächter, and T. Asfour, "Learning object-action relations from bimanual human demonstration using graph networks," *IEEE RA-L*, vol. 5, no. 1, pp. 187–194, 2019. https://bimanual-actions.humanoids.kit.edu/.
44. E. Özsoy, E. P. Örnek, U. Eck, T. Czempiel, F. Tombari, and N. Navab, "4d-or: Semantic scene graphs for or domain modeling," *arXiv preprint arXiv:2203.11937*, 2022. https://github.com/egeozsoy/4D-OR.
45. A. Goyal, K. Yang, D. Yang, and J. Deng, "Rel3d: A minimally contrastive benchmark for grounding spatial relations in 3d," *NeurIPS*, vol. 33, pp. 10514–10525, 2020. https://github.com/princeton-vl/Rel3D.
46. Y. Hong, L. Yi, J. Tenenbaum, A. Torralba, and C. Gan, "Ptr: A benchmark for part-based conceptual, relational, and physical reasoning," *NeurIPS*, vol. 34, pp. 17427–17440, 2021. http://ptr.csail.mit.edu/.
47. J. Wald, A. Avetisyan, N. Navab, F. Tombari, and M. Nießner, "Rio: 3d object instance re-localization in changing indoor environments," in *CVPR*, pp. 7658–7667, 2019. https://waldjohannau.github.io/RIO.
48. F. Xia, A. R. Zamir, Z.-Y. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: real-world perception for embodied agents," in *CVPR*, 2018. http://gibsonenv.stanford.edu/.
49. K. Tang, H. Zhang, B. Wu, W. Luo, and W. Liu, "Learning to compose dynamic tree structures for visual contexts," in *CVPR*, June 2019.
50. N. Gkanatsios, V. Pitsikalis, P. Koutras, and P. Maragos, "Attention-translation-relation network for scalable scene graph generation," in *ICCV*, Oct 2019.
51. X. Li, D. Guo, H. Liu, and F. Sun, "Embodied semantic scene graph generation," in *CoRL*, vol. 164 of *Proceedings of Machine Learning Research*, pp. 1585–1594, PMLR, 2022.
52. F. Wu, F. Yan, W. Shi, and Z. Zhou, "3d scene graph prediction from point clouds," *Virtual Reality & Intelligent Hardware*, vol. 4, no. 1, pp. 76–88, 2022.
53. C. Agia, K. M. Jatavallabhula, M. Khodeir, O. Miksik, V. Vineet, M. Mukadam, L. Paull, and F. Shkurti, "Taskography: Evaluating robot task planning over large 3d scene graphs," in *CoRL*, pp. 46–58, 2022.
54. Z. Jiao, Y. Niu, Z. Zhang, S.-C. Zhu, Y. Zhu, and H. Liu, "Sequential manipulation planning on scene graph," in *IROS*, 2022.
55. Z. Ravichandran, L. Peng, N. Hughes, J. D. Griffith, and L. Carlone, "Hierarchical representations and explicit memory: Learning effective navigation policies on 3d scene graphs using graph neural networks," in *ICRA*, pp. 9272–9279, 2022.
56. H. Dhamo, F. Manhardt, N. Navab, and F. Tombari, "Graph-to-3d: End-to-end generation and manipulation of 3d scenes using scene graphs," in *CVPR*, pp. 16352–16361, 2021.
57. A. Savkin, R. Ellouze, N. Navab, and F. Tombari, "Unsupervised traffic scene generation with synthetic 3d scene graphs," in *IROS*, pp. 1229–1235, IEEE, 2021.