Open3DSG: Open-Vocabulary 3D Scene Graphs from Point Clouds with Queryable Objects and Open-Set Relationships – Supplementary Material –

Sebastian Koch[1,2,3] Narunas Vaskevicius[1,2] Mirco Colosi[2]
Pedro Hermosilla[4] Timo Ropinski[3]
[1]Bosch Center for Artificial Intelligence [2]Robert Bosch Corporate Research [3]University of Ulm [4] TU Vienna
kochsebastian.com/open3dsg

In this **supplementary material**, we first provide additional implementation details in Sec. A. Next, we detail our design choices for our open-vocabulary 3D scene graph approach in Sec. B. In Sec. C we provide additional details on our proposed baselines. Next, we highlight the improved semantic understanding of our open-vocabulary method compared to fully-supervised methods in Sec. D and demonstrate the advantages with long-distance relationships compared to 2D-only open-vocabulary methods in Sec. E. We show unique applications of how our open-vocabulary 3D scene graphs can be used in Sec. F. Finally, we provide more qualitative results in Sec. G.

A. Implementation details

For our 3D graph backbone, we extract features from the point cloud using two PointNets that compute an initial 1024-dimensional feature vector for each node and edge. The graph features are refined using five layers of graph convolutions with message passing inspired by [6] and a hidden dimension of 2048. Finally, the node features are projected into the 768-dimensional CLIP space using a 5-layer MLP with ReLU activations and batch norm. The edge features are concatenated with the positional encoding from the BLIP-ViT and projected into the 1408-dimensional BLIP feature space using a 5-layer transformer architecture. The model is trained for 50 epochs using the Adam optimizer with weight decay, a learning rate of 5e-4, and a cyclic cosine-annealing learning rate scheduler. We use a batch size of 6 on a single Nvidia A100 GPU with mixed-precision.

During inference time, we use the pre-trained CLIP ViT-L/14@336 text encoder to encode the object queries and a pre-trained Vicuna 7B LLM model from Hugging Face [1] for predicate prediction. To query the CLIP text encoder we use object classes from the 160 class label set from 3DSSG [6], but we are not limited to those and can also query other arbitrary object classes or even concepts rather than discrete classes. To prompt the LLM we design an open-ended prompt to get the most open-vocabulary response: "Describe the relationship between [*object1*] and [*object2*]?". Here *object1* and *object2* are the object classes queried in the first step by CLIP. It is also possible to ask whether a specific relationship exists. However, we observe that providing more than five options confuses the LLM. To map the LLM predictions to the closed-vocabulary benchmark label set, we use the bert-base-uncased model from Hugging Face [2] with 768-dimensional feature embeddings.

B. Design choices

To succeed with distilling an open-vocabulary 3D scene graph method from 2D foundation models, we first study which model and which dataset is best suited for the distillation.

Compositionality pilot-study. Our approach highly depends on the knowledge encoded in the 2D vision-language model. However, Yuksekgonul et al. [8] and others [7] have demonstrated that current contrastive pre-trained vision-language models behave like bag-of-words models and have little understanding of compositionality. To evaluate whether a contrastively pre-trained VLM is suited for the distillation into our 3D scene graph model, we perform a pilot-study on a subset of the VL-Checklist Relation [8] benchmark. Differently from the evaluations conducted in [8], we do not evaluate whether the VLM can differentiate between the correct and incorrect relationship description but provide a set of queries where the VLM has to choose the most likely. This makes the task much harder for the VLM as the likelihood that the VLM picks the correct caption among the incorrect captions by random chance is much smaller. In the evaluation, we query the VLM using the query template "A relationship of a [*subject*] is [*predicate*] a [*object*]", where

[1] https://huggingface.co/Salesforce/instructblip-vicuna-7b

[2] https://huggingface.co/bert-base-uncased

subject and *object* are fixed to the ground truth to solely evaluate the relationship understanding of the VLM. We report the top-1, top-2, and top-5 recall scores denoting whether the correct predicate was in the top-k highest similarity scores.

	top-1	top-3	top-5
Random chance	0.04	0.12	0.19
CLIP (ViT-L/14)	0.12	0.30	0.42
NegCLIP	0.14	0.35	0.48
SigLIP	0.11	0.27	0.37

Table A. **VL-Checklist Relation.** We evaluate the embedded relationship knowledge of the current state of contrastively pre-trained VLMs on an adapted benchmark from [52]. Results are reported for whether the VLM scores the correct predicate in the top-1, top-3, or top-5.

As expected, while CLIP [5], NegCLIP [8], and SigLIP [9] are exceptional zero-shot classifiers of objects, they cannot model inter-object relationships. The experimental evidence on a small controlled evaluation benchmark indicates that CLIP-like contrastively pre-trained VLMs do not have enough compositional knowledge about relationships that can be distilled into a 3D network. Therefore, in this paper, we choose to go beyond CLIP-like VLMs for relationship prediction and leverage a BLIP [2] vision encoder that can be projected into the token space of an LLM via a Qformer to predict relationships.

Distillation dataset. We choose to distill features on ScanNet [1] rather than 3RScan / 3DSSG [6], which we evaluate on. The reason for this is highlighted in Fig. A. Both datasets are indoor datasets depicting similar scenes. While ScanNet was recorded with an iPad with an attached depth sensor in landscape mode, 3RScan / 3DSSG was recorded with a Google Tango in portrait mode. The different recording setups result in entirely different vertical and horizontal field-of-views. We reason that to extract meaningful visual features representing the relationships between two objects, it is necessary that two objects are nearly fully visible in the same frame. This is rarely the case in 3RScan with its portrait setup. Therefore, we choose to use ScanNet for distillation as more of its frames depict more than one object.

C. Baselines

In addition to proposing a novel open-vocabulary 3D scene graph prediction method, we also propose several baselines. Here we provide further details on these baselines.

CLIP (naive). The most naive approach is to predict objects and predicates independently from each other directly using CLIP [5]. We select images for each object instance as well as images where a pair of objects is shown similar to the process in Sec. 3.2 and encode them using the CLIP image encoder. Then we build a fully-connected graph from the encoded features and query the nodes with object class labels and the edges with predicate class labels.

3RScan / 3DSSG

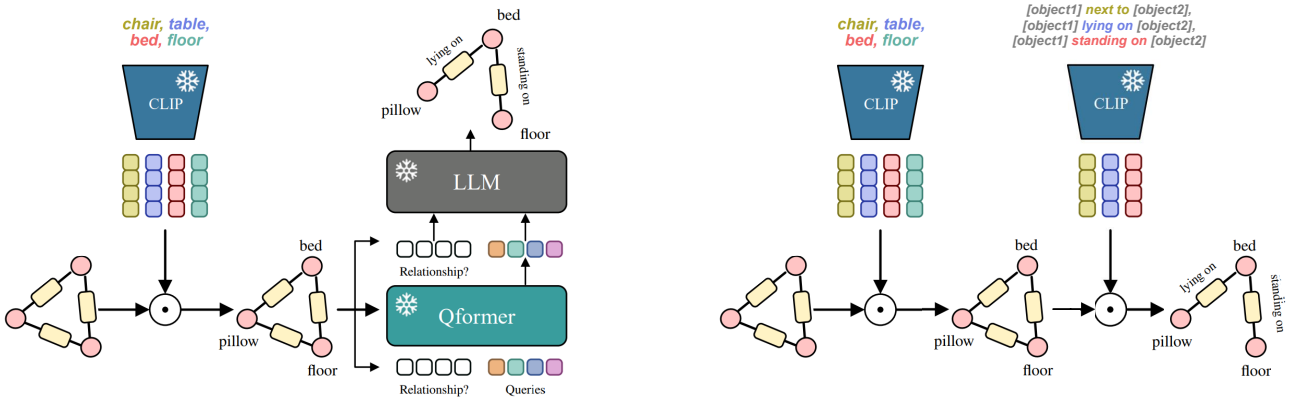
ScanNet

Figure A. **ScanNet vs. 3RScan.** We choose ScanNet over 3RScan / 3DSSG as a distillation dataset since the FOV of each frame is generally higher and more objects are visible in one frame.

CLIP & NegCLIP. A more sophisticated approach using CLIP [5] or NegCLIP [8] is more similar to our two-step approach. The difference is shown in Fig. B. Here we also first build a fully-connected feature graph and predict object classes by querying the class of each node. Then we use the predicted objects as context to query full relationships in a second step using CLIP. Using the predicted objects as context improves results compared to the naive approach, nevertheless, the results fall short of our LLM approach due to the limited compositional knowledge of both CLIP and NegCLIP.

D. Improved semantics

While Tab. 1 in the main paper shows that our proposed open-vocabulary 3D scene graph method achieves overall worse performance compared to the current SOTA fully-supervised methods, Tab. 2 demonstrates the advantages of an open-vocabulary method, where we outperform the fully-supervised baselines on long-tail distribution classes. To give further insights into the benefits of our proposed



(a) Generative: LLM-based Relationships (Ours) **(b) Discriminative: CLIP-based Relationships (Baseline)**

Figure B. **Relationship prediction comparison.** We compare our generative relationship prediction approach using a prompted LLM (a), with a CLIP-based querying baseline (b) from Tab. 1. Due to the limited compositional knowledge of CLIP-like models, a discriminative approach where predicates can be directly queried performs much worse than a generative LLM-based approach.

open-vocabulary method, we provide scores on selected object and predicate classes in Tab. B. It shows that our open-vocabulary method outperforms the fully-supervised methods on very specific and semantically descriptive classes. For instance, for objects our network is better at differentiating a *chair* from a *dining chair* or a *table* from a *bedside table*. At the same time, fully-supervised methods, likely due to class imbalance during training, often only predict a generic class rather than the most specific class possible. This is similar for predicates. While the fully-supervised methods generally perform well on all predicates, highly semantic and specific predicates such as *covering* or *belonging to* are predicted less accurately. In contrast, our open-vocabulary method performs particularly well on semantic predicates such as *standing on*, *covering* or *belonging to*.

	3DSSG	SGRec3D	Open3DSG
Objects R@5			
cabinet / kitchen cabinet	0.39 / 0.33	0.67 / 0.87	0.39 / 0.94
chair / dining chair	0.98 / 0.00	0.94 / 0.00	0.48 / 1.00
table / bedside table	0.60 / 0.00	0.90 / 0.25	0.37 / 1.00
Predicates R@3			
standing on	0.73	0.95	0.86
covering	0.00	0.00	0.24
belonging to	0.48	0.65	0.91

Table B. **Semantic awareness.** While fully-supervised methods such as 3DSSG [44] and SGRec3D [22] produce overall good results, their performance on difficult, rare, and semantically descriptive classes remains low. In contrast our open-vocabulary approach excels at semantically descriptive classes.

E. Long distance relationships

In Tab. 3, we provide an ablation for 3D scene graph prediction solely with 2D vision-language models. Only using 2D data performs worse than our learned 2D-3D ensemble approach.

While a prediction using 2D images is possible, a significant disadvantage of relying only on 2D data is that to predict a relationship between two objects, those two objects must be visible together in at least one frame. In contrast, our method does not have this limitation since it processes the 3D point cloud and can predict a relationship between two objects of arbitrary distance in a point cloud. Fig. C shows such two far-apart objects that are not close enough to appear in a shared frame, but still have a meaningful relationship detected by our method.

F. Applications

F.1. 3D Triplet localization

3D scene graphs are useful for various downstream computer vision or robotics tasks. In Fig. D we demonstrate one of those use cases uniquely suited to our language-aligned open-vocabulary 3D scene graphs. First, a 3D scene is encoded as an open-vocabulary 3D scene graph using our method. This representation is now queryable and promptable with an open vocabulary, making it a versatile tool for various scene understanding tasks. We demonstrate its usefulness for object localization in a 3D point cloud. Unlike other object localization methods [4], our goal is not to localize all objects of the same class but a specific instance that fits a relationship description. We encode a relationship description using the CLIP [5] and BERT [3] language encoders to generate a triplet feature representing the relation-

Figure C. **Long distance relationships.** In contrast to a 2D-only relationship prediction approach, which requires two objects to be visible in an image together, our 3D approach can predict relationships for two arbitrary far objects.

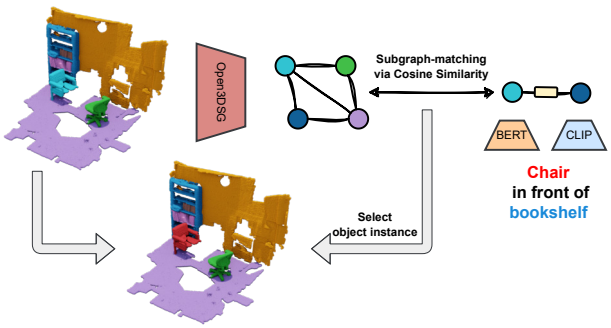


Figure D. **Application: Object localization via triplet description.** Using our open-vocabulary approach, we can localize object instances in the 3D point cloud given a relationship description of the object instance.

ship. Then, we perform a subgraph-matching based on the cosine similarity of each triplet in the encoded scene graph with our target triplet feature. We select the triplet with the highest similarity score and reference it in the point cloud using the scene graph-point cloud alignment.

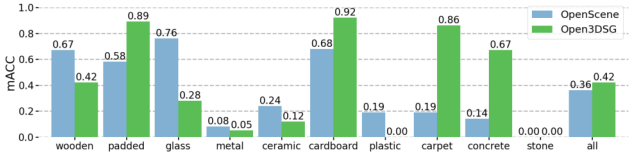


Figure E. **Application: Material prediction.** Using our open-vocabulary approach, we predict the material of objects without explicit training. We compare against OpenScene [4].

F.2. Material prediction

We present another application of zero-shot object attribute/material prediction, evaluated quantitatively in Fig. E. The material prediction can be performed without further training with the same querying strategy described in Sec. 3.4. Predicting attributes for each object further enriches the predicted 3D scene graph. We provide a top-1 accuracy metric comparison with OpenScene [4], a point cloud-based open-vocabulary method, on 3DSSG. Open3DSG outperforms OpenScene for most materials and also achieves a higher average accuracy for all classes. Note however that OpenScene predicts the material per point while we predict the material per instance.

F.3. Reasoning over object affordances

A further application is the reasoning over scene-specific affordances using Open3DSG. Given the open-vocabulary representation computed by our method, we can prompt the LLM to predict affordances between objects. These affordances are grounded by the processed scene. In Fig. F, we demonstrate how Open3DSG can reason over which objects can be picked up by a human by prompting the LLM "Can you lift [x] from [y]". Our model correctly predicts that the pillows can be picked up from the bed while the bed would be too heavily to lift from the carpet.

G. Additional 3D scene graph predictions

In Fig. G, we provide additional 3D scene graph predictions on ScanNet [1]. Relationships for objects that are further apart than 0.5m are pruned for clarity in the visualization. Overall, the 3D scene graph predictions are correct and the advantages of an open-vocabulary method become especially apparent for rare and specific object classes such as *computer desk* or precise relationship descriptions such as *tv mounted on wall*. But our open-vocabulary approach still has several limitations, such as overall low diversity in the predicted relationships. However, this limitation is not unique to our open-vocabulary method but also remains an issue with the current state of fully-supervised methods.

Nonetheless, our approach also has unique limitations, such as LLM-typical hallucinations like *computer desk (keyboard) connected via USB to monitor* or imperfect geometric

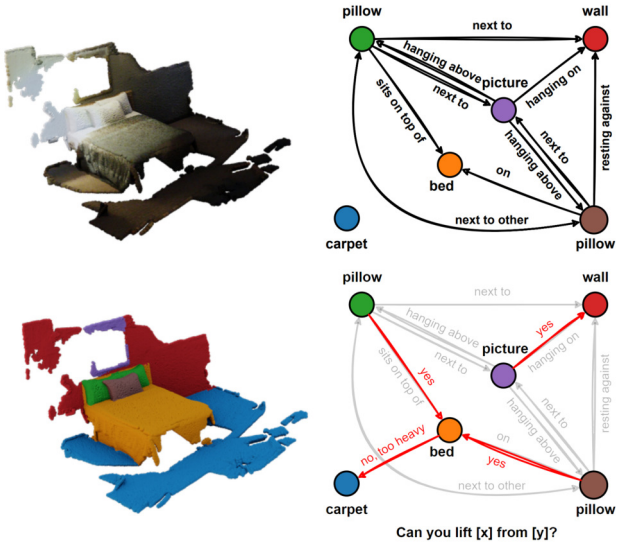


Figure F. **Application: Reasoning over object affordances.** Using our open-vocabulary approach, we reason about the affordances of objects by for instance prompting the LLM to output whether an object can be lifted from the other.

understanding where two monitors are both predicted to be *to the left of* each other.

References

[1] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5828–5839, 2017. 2, 4

[2] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 2

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[4] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3, 4

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 3

[6] Johanna Wald, Helisa Dhamo, Nassir Navab, and Federico Tombari. Learning 3d semantic scene graphs from 3d indoor reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2

[7] Yutaro Yamada, Yingtian Tang, and Ilker Yildirim. When are lemons purple? the concept association bias of clip. *arXiv preprint arXiv:2212.12043*, 2022. 1

[8] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *International Conference on Learning Representations*, 2023. 1, 2

[9] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *arXiv preprint arXiv:2303.15343*, 2023. 2

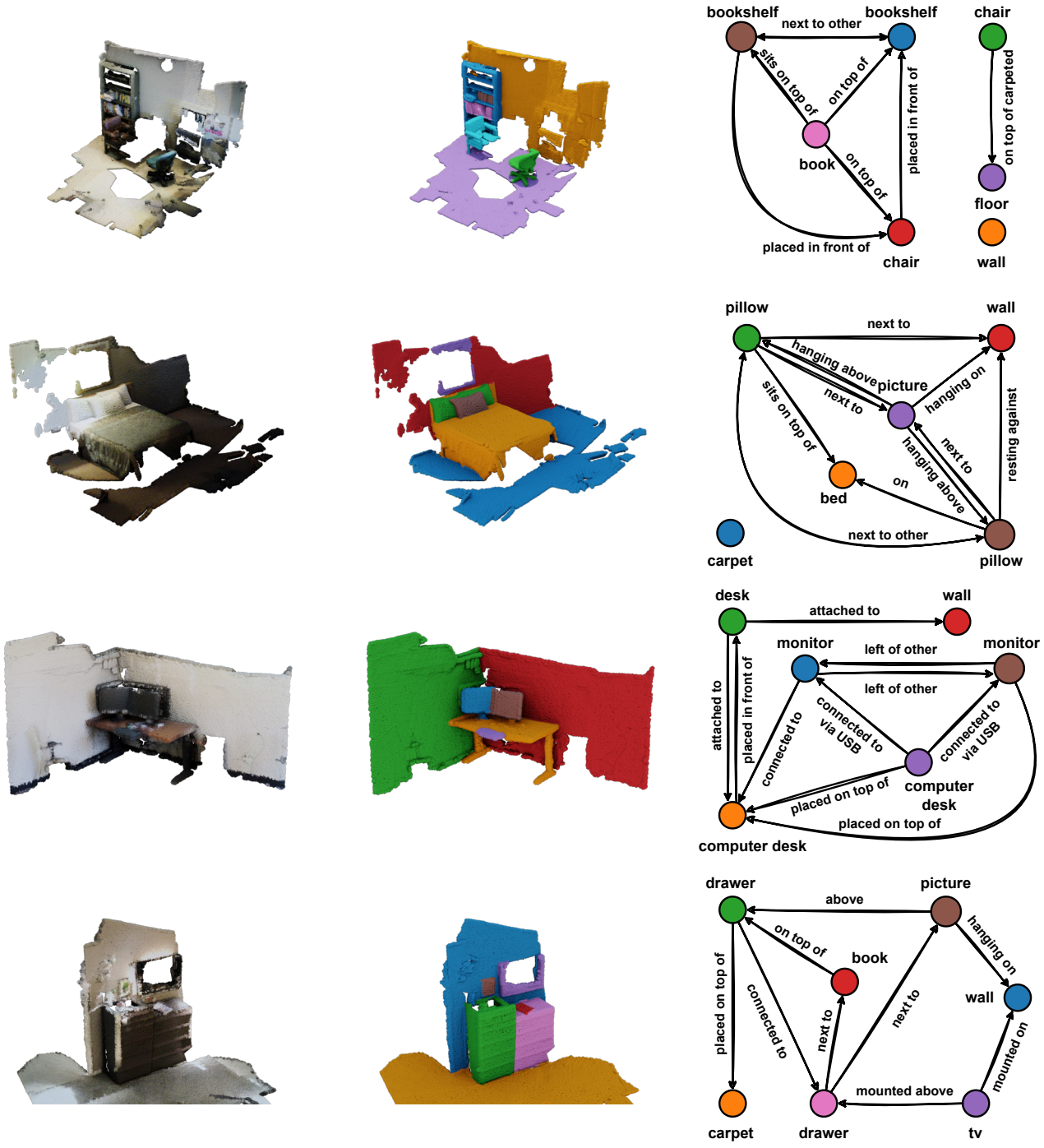


Figure G. **Qualitative open-vocabulary 3D scene graph predictions.** Left: Colored point cloud input; Middle: Class-agnostic mask; Right: Predicted open-vocabulary 3D scene graph.