ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Autonomous Systems Lab
Prof. Roland Siegwart

Semester Thesis

Incremental hierarchical 3D scene graph construction for high-level planning

Spring Term 2021

Supervised by:
Lukas Schmid
Julian Foerster

Author:
Junting Chen

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

INCREMENTAL HIERARCHICAL 3D SCENE GRAPH CONSTRUCTION FOR HIGH-LEVEL PLANNING

Authored by (in block letters):

For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
CHEN	JIANTING

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**
July 22, 2021	J.C

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.

Contents

Abstract

In this paper, we introduce a system to construct a hierarchical 3D scene graph from dense mapping for high-level planning incrementally in real time. The proposed system leverages both the geometric information and hierarchical space structure of the environment and rich inter-objects semantic relationships. Besides, we demonstrate the idea of aggregation of semantic information by predicting the room class from the contained objects. Finally, we evaluate this system with our own multi-room environment, generated by photo-realistic simulator.

Chapter 1

Introduction

The progress of SLAM-based environment reconstruction, learning-based environment understanding and capability of robots' sensing and computation in recent years draws much attention to the understanding and high-level planning ability of robots in complex indoor environment, in both robotics community and computer vision community. In this scenario, the term Embodied Artificial Intelligence [1] is proposed to refer to those tasks involving vision, languages, actions and control with embodied robots running in either real world or virtual environment. Many Embodied AI tasks, including Embodied Question Answering [2], Visual Navigation [3], Visual Language Navigation [4], have become hot research topics in interdisciplinary communities.

One critical problem in embodied AI is how to learn and represent environment knowledge, including unary and binary, geometric and semantic, transient and long-term information about the environment. Thanks to the strong expressiveness of graph data structure and the recent success of Graph Convolutional Networks [5] in many fields, more and more work focus to extract 3D scene graphs from indoor environment[6][7][8][9]. Previous works either focused on the geometric information and layered structure of the environment, or only on the semantic relationships extracted by learning-based models. Besides, most of them build the 3D scene graph after the semantic mesh is constructed, by offline analysis. In this paper, we present a system to extract both hierarchical 3D scene graph with 1) Inter-objects semantic relationships, predicted from object point clouds 2) room segmentation, generated from TSDF voxels 3) room class, predicted from objects in rooms. Finally, we evaluate our system on a multi-room scene generated from photo-realistic simulator and discuss its possible usage in high-level planning tasks.

Chapter 2

Related Works

Scene Graph [10], defined as "a structured representation of a scene that can clearly express the objects, attributes, and relationships between objects in the scene" in a recent survey paper [11], are a popular representation in many Vision-Language multimodal tasks. In Visual Question Answering (VQA), given an image and several questions concerning the scene in image, the model is expected to answer a question written in natural language. Many previous works [12] [13] [14] try to perform explicit reasoning on the scene graph with detected objects as graph nodes. In Visual Dialogue Generation, this work [15] present a dynamic scene graph representation learning pipeline that consists of an intra-frame reasoning layer and an inter-frame aggregation module capturing temporal cues to generate meaningful dialogue.

3D Scene Graph [7], instead of describing a scene or an event described by a 2D image or a segment of video, a 3D scene graph describes an environment (usually indoor) including objects, properties, inter-object relations. Transferring from 2D data, such as images and videos, to 3D data like RGBD point clouds and reconstructed meshes from SLAM-based dense mapping, many works [6][16][8] naturally try to use end-to-end deep neural models to predict the 3D scene graph, as end-to-end models have achieved great success in 2D data. While [16] focuses on detecting objects and predicting inter-objects relations from the video clip of exploring indoor environment, [8] predicts object class and inter-objects relations on segmented RGB point clouds, generated by SLAM-based dense mapping. Due to the poor performance of end-to-end models, some other works [7][9] combine the automated pipeline to gather geometric information and create ground-truth annotations on semantic information including properties and inter-objects relations by human labor. [9] also proposes a "full-stack solution" to indoor environment understanding, which incorporates state-of-the-art SLAM framework Kimera [17], with pose graph estimation module, to provide low-level visual information. Then the following pipelines performs offline processing to extract semantic information from dense reconstruction, including object classes, properties, inter-object relationships, places and structures of the room, etc. These works construct 3D scene graph by data analysis, after the SLAM-based dense mapping finishes and overall environment reconstruction is ready. Instead, our proposed system focuses on 3D scene graph construction in the real time.

There are also some works already trying to utilize 3D scene graph as explicit knowledge representation to help finish high-level planning tasks in the environment. [18] proposes to use a 3D scene graph representation to help robots look for objects in indoor environment.

Chapter 3

Method

In this chapter, we firstly present the overall pipeline to generate the hierarchical 3D scene graph, and then give a detailed description of our target 3D scene graph, followed by detailed explanations of each of the module in the system. Figure 3.1 gives the overview of the proposed pipeline.

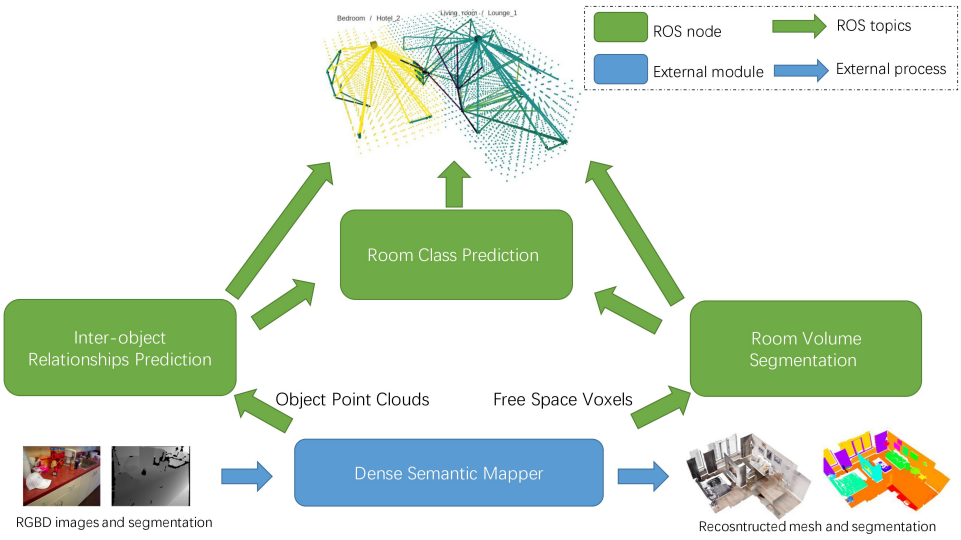
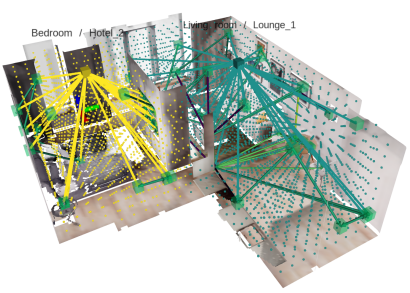


Figure 3.1: Overview of the pipeline to generate hierarchical 3D scene graph

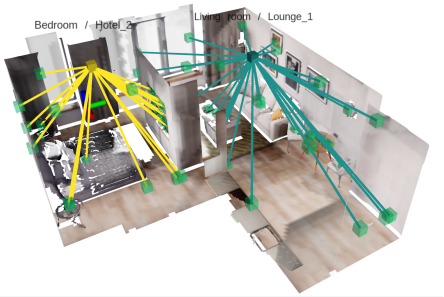
As can be seen in figure 3.1, taking a stream of RGBD images and semantic segmentations as input, the dense semantic mapper (blue box at the bottom) reconstruct the colored mesh and mesh segmentation using TSDF-based method [19]. On top of this dense mapping pipeline, we introduce three extra modules to extract the 3D scene graph: 1) Inter-object Relationships Prediction Module 2) Room Volume Segmentation Module 3) Room Class Prediction Module, visualized as green boxes in the middle of figure 3.1. These three modules take into point clouds of objects and voxels of 3D free space as input, and generate the hierarchical 3D scene graph as depicted on the top of figure 3.1.

3.1 Hierarchical 3D Scene Graph

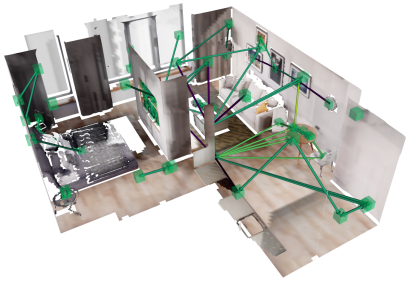
Our hierarchical 3D scene graph consists of two layers: 1) object layer 2) room layer. With RGB point clouds of object and voxels of free space mapping generated from the provided exterior dense mapper, the system generates 1) inter-object relationships 2) room segmentation in voxels 3) room class labels. The demo result on a two-room flat is shown in figure 3.2a.



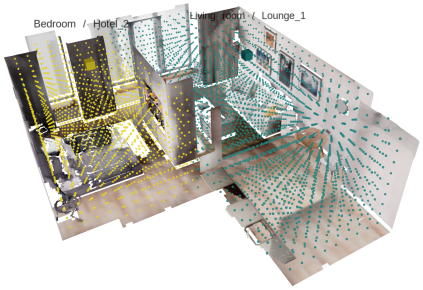
(a) Overall result



(b) Objects and rooms



(c) Inter-objects relations



(d) 3D Room Segmentation

Figure 3.2: Visualization: Generated hierarchical 3D scene graph

In figure 3.2b, we only visualize the backbone of the 3D scene graph:

- Object nodes: Green cubes. The position of each cube represents the centroid of point clouds of the corresponding object.
- Room nodes: Colored cubes over green cubes. The position of each cube represents the centroid of free space voxels belonging to the room.
- "Is in" relationships: Colored edges that connect all object nodes to one of the room nodes.

These three elements represent the natural spatial hierarchy of "something is in some place".

Figure 3.2c only shows the semantic relationships between objects in the environment, with different colors representing different semantic labels.

Figure 3.2d visualizes the free space voxels segmented to room space. The grid of spheres are the centers of free space voxels. Each voxel belongs to the room node with the same color.

3.2 Inter-object Relationships Prediction

This module takes point clouds of objects as input, provided by the dense semantic mapper, and generate inter-object relationships. We notate the input object RGB point clouds as $P = \{P_1, P_2, \cdots, P_i, \cdots, P_N\}$, where $P_i \in \mathbb{R}^{K_i \times 6}$ represents K_i points of i-th object. Each point is a vector of shape $p_i^j = [x_i^j, y_i^j, z_i^j, r_i^j, g_i^j, b_i^j]$, where $[x_i^j, y_i^j, z_i^j]$ are the coordinates of the point, and $[r_i^j, g_i^j, b_i^j]$ are the color values in RGB color space. Then the vertex normal of each point is calculated by finding adjacent points and calculating the principal axis of the adjacent points using covariance analysis. The adjacent points of the target point are searched in a sphere of radius=0.1, with the target point at the center. We concatenate estimated vertex normals and original point clouds as the processed point clouds, $P' = \{P'_1, P'_2, \cdots, P'_i, \cdots, P'_N\}$, where $P_i \in \mathbb{R}^{K_i \times 9}, p_i^{'j} = [x_i^j, y_i^j, z_i^j, r_i^j, g_i^j, b_i^j, nx_i^j, ny_i^j, nz_i^j]$, $[nx_i^j, ny_i^j, nz_i^j]$ are estimated vertex normals.

Then we use the processed point clouds to predict inter-object relationships. In this module, we use pre-trained model of 3DSSG [8], which takes into the point cloud points sampled to $M = 256$, of the subject and object of the relationship, along with a edge descriptor, to predict the relationship class label. An edge descriptor is a 11-dim vector composed of following values: given the target objects i, j, the relationship $i- > j$ is to be predicted, and the edge descriptor, composed of:

- Offset between the centroids of two target object i, j, $[x_i, y_i, z_i] - [x_i, y_i, z_i]$.
- Difference of standard deviations of point clouds positions on (x,y,z) axes, $[std_{x,i}, std_{y,i}, std_{z,i}] - [std_{x,j}, std_{y,j}, std_{z,j}]$.
- Logarithm ratio of 3D dimensions of the two target objects i, j, $\log([dim_{x,i}, dim_{y,i}, dim_{z,i}]/[dim_{x,j}, dim_{y,j}, dim_{z,j}])$.
- Logarithm ratio of the two target objects' volumes, $\log(v_i/v_j)$.
- Logarithm ratio of the two target objects' lengths (largest dimension), $\log(l_i/l_j)$.

An edge descriptor is the concatenation of all five values mentioned above, which is then used as the initial value of edge node in GCN.

For more details about the implementation of 3DSSG model, please refer to [8].

3.3 Room Volume Segmentation

This module predicts the room segmentation over free space voxels, generated by the dense mapper, notated as $V = [v_1, v_2, \cdots, v_i, \cdots, v_M]$, where $v_i \in \mathbb{R}^3$ represents the center one voxel in space. The predefined voxel size is $d = 0.25m$, and thus the distance between two adjacent voxel centers on the grid is also $d = 0.25$. There are many choices of methods to segment room space, and here we choose a intuitive and simple method: 1) Erode the free space voxels 2) Detect and assign IDs to each of the connected sub-component in the remaining free space voxels 3) Dilate/Propagate the assigned IDs to neighboring unassigned free space voxels until all free space voxels are assigned with IDs.
For the erosion, we choose the erosion margin $1.25m$, or equivalently, 5 voxel centers, which is about the size of a door in a common indoor environment. In disconnected component

3.4 Room Class Prediction

Humans are good at inference: we can easily predict what the room is used for, from the furniture, decorations, and other objects in that room. After room segmentation, we now assign each object to its nearest room node, as shown in figure 3.2b. Then we extract the simplest feature of objects in one room, that is, the number of occurrences for each object class. We annotate occurrence vectors in a scene as $O = [O_1, O_2, \cdots, O_r, \cdots, O_R], O_r \in \mathbb{R}^C$, where R, C represent the number of rooms in the segmentation and pre-defined number of object classes in the dataset, respectively. The c-th value in vector O_r, written as O_r^c, is the number of objects of class c occurring in room r.

With occurrence vectors, a simple random forest classifier is used to predict the room class label. Due to the lack of annotated scenes for this project, we train the classifier on a large-scale public indoor scan dataset, ScanNet [20]. ScanNet contains more than 1500 indoor room scans, with instance-level annotations and scan labels.

Chapter 4

Evaluation

Since the inter-objects relationships are predicted by the pre-trained 3DSSG [8] model, we only give the brief evaluation result about the room segmentation and room classification.

Since many photo-realistic indoor scans dataset, including ScanNet [20] and 3RScan [21], have their scans already been segmented into single rooms, we annotate one multi-room indoor scan generated in a photo-realistic simulator, as shown in figure 4.1, to evaluate our result.

Figure 4.1: Dense Reconstruction of the Multi-room Scan

Figure 4.2 gives the qualitative result of the room segmentation and room class prediction on the introduced multi-room scan. The segmentation at the top is the annotated ground truth while the segmentation at the bottom is our prediction. The most obvious difference is that, our prediction cannot recognizes the "Hallway" area. This indicates one possible drawback of our method to predict room class, that is, our method heavily depends on the semantics of the objects in the area. Since "Hallway" is a notion that is decided by the shape of the area and its relative position to connect multiple rooms, our object-based method naturally fails in this case.

Figure 4.3 shows how 1) number of rooms 2) IOU (intersection over union) 3) room class prediction accuracy change with respect to running time. For each predicted segment, IOU is calculated by selecting the maximum IOU score with all ground truth segments. The IOU score shown in figure 4.3b is averaged over all predicted segments. For room class prediction, we match a predicted segment to the ground truth segment with maximum IOU score, and calculate the prediction accuracy be-

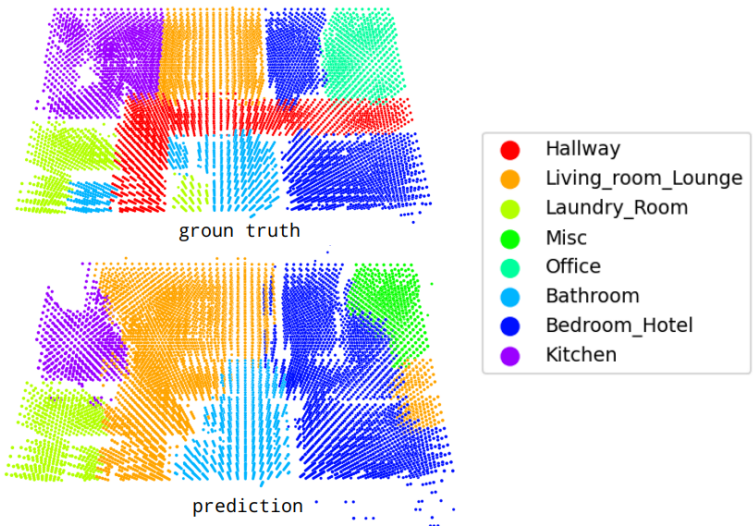
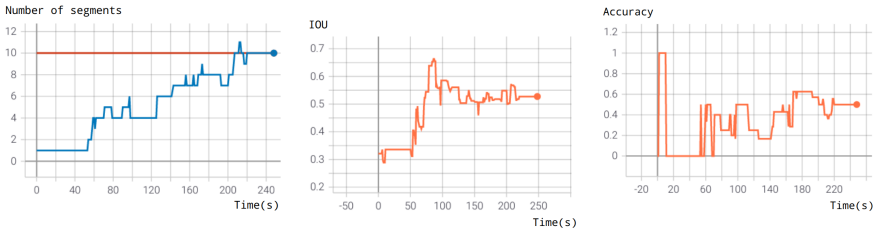


Figure 4.2: Visualization of the Room Segmentation and Prediction Result

tween matched pairs.



(a) Number of Room Segments

(b) Segmentation IOU

(c) Room Prediction Accuracy

Figure 4.3: Metric Score Change w.r.t. Running Time of System

Chapter 5

Conclusion

In this paper, we present a system to incrementally construct hierarchical 3D scene graph built on top of a dense mapper. This system predicts inter-object relations from object point clouds, creates room segmentation and predicts room class in the real time, rather than perform offline data analysis after dense reconstruction is finished, compared with previous works.

However, this system is composed of basic algorithms and easy aggregation strategy, and thus has much potential of improvement in many aspects. Here are some directions that are worth trying to improve the performance of the system:

- For inter-objects relationships prediction, now only 3D point clouds of objects are used. However, the quality of mesh reconstruction does constrain the theoretically best possible performance of the present model. Besides, 2D images provide some extra visual information like texture of objects that could be useful for relation prediction. For example, objects made of hard material is more probable than soft objects to "support to" some other objects. Thus, it is worth trying to fuse the inter-objects relation predicted from 3D point clouds and 2D video stream.
- For room segmentation, now only a simple "Erosion-Dilation" algorithm is applied. However, there are many heuristics in indoor area segmentation, such as walls, which are natural borders of different areas in a scene. These visual heuristics could help to improve the segmentation performance.
- For room prediction, not only the objects in an area that counts, but also the shape of the area and the semantics of the neighboring areas.

After all, 3D scene graph is a structured middle representation between semantic space and visual space, and it also enables explicit reasoning process on data structure that describes the environment. Thus, 3D scene graph provides a seemingly viable way to answer the question, how to perform high-level planning tasks that might involve semantics, visual understanding, actions and control. After we can construct a "good enough" 3D scene graph, combine it with high-level planning is naturally the step further.

Bibliography

[1] M. Hoffmann and R. Pfeifer, "The implications of embodiment for behavior and cognition: animal and robotic case studies," *CoRR*, vol. abs/1202.0440, 2012.

[2] E. Wijmans, S. Datta, O. Maksymets, A. Das, G. Gkioxari, S. Lee, I. Essa, D. Parikh, and D. Batra, "Embodied Question Answering in Photorealistic Environments with Point Cloud Perception," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[3] S. Gupta, J. Davidson, S. Levine, R. Sukthankar, and J. Malik, "Cognitive mapping and planning for visual navigation," *CoRR*, vol. abs/1702.03920, 2017.

[4] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. D. Reid, S. Gould, and A. van den Hengel, "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," *CoRR*, vol. abs/1711.07280, 2017.

[5] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[6] U.-H. Kim, J.-M. Park, T.-J. Song, and J.-H. Kim, "3d-scene-graph: A sparse and semantic representation of physical environments for intelligent agents," *IEEE Cybernetics*, 2019.

[7] 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 *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[8] J. Wald, H. Dhamo, N. Navab, and F. Tombari, "Learning 3d semantic scene graphs from 3d indoor reconstructions," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[9] A. Rosinol, A. Gupta, M. Abate, J. Shi, and L. Carlone, "3d dynamic scene graphs: Actionable spatial perception with places, objects, and humans," *arXiv preprint arXiv:2002.06289*, 2020.

[10] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei, "Image retrieval using scene graphs," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3668–3678.

[11] P. Xu, X. Chang, L. Guo, P.-Y. Huang, X. Chen, and A. Hauptmann, "A survey of scene graph: Generation and application," 2020.

[12] D. Teney, L. Liu, and A. V. Hengel, "Graph-structured representations for visual question answering," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3233–3241, 2017.

[13] J. Shi, H. Zhang, and J.-Z. Li, "Explainable and explicit visual reasoning over scene graphs," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8368–8376, 2019.

[14] D. A. Hudson and C. D. Manning, "Learning by abstraction: The neural state machine," in *NeurIPS*, 2019.

[15] S. Geng, P. Gao, M. Chatterjee, C. Hori, J. LeRoux, Y. Zhang, H. Li, and A. Cherian, "Dynamic graph representation learning for video dialog via multi-modal shuffled transformers," in *Proceedings of the Thirty-Fifth Conference on Association for the Advancement of Artificial Intelligence (AAAI)*, 2021, pp. 1415–1423.

[16] P. Gay, S. James, and A. Del Bue, "Visual graphs from motion (vgfm): Scene understanding with object geometry reasoning," 2018.

[17] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2020.

[18] A. Kurenkov, R. Martín-Martín, J. Ichnowski, K. Goldberg, and S. Savarese, "Semantic and geometric modeling with neural message passing in 3d scene graphs for hierarchical mechanical search," *arXiv preprint arXiv:2008.07792*, 2020.

[19] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[20] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.

[21] J. Wald, A. Avetisyan, N. Navab, F. Tombari, and M. Niessner, "Rio: 3d object instance re-localization in changing indoor environments," 2019.