

QueSTMaps: Queryable Semantic Topological Maps for 3D Scene Understanding

Yash Mehan^{1*}, Kumaraditya Gupta^{1*}, Rohit Jayanti^{1*}, Anirudh Govil¹, Sourav Garg², and Madhava Krishna¹

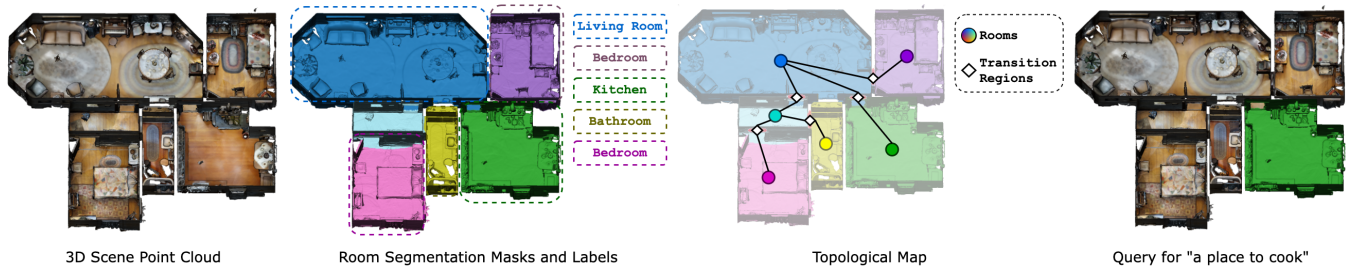


Fig. 1: **Topology Extraction and Room Labeling for Indoor Scene 3D Point clouds.** Given a 3D scene point cloud we (a) predict room and transition regions, (b) generate room-label aligned CLIP embeddings to assign room labels, and (c) build a topological map that supports room-level natural language queries, e.g., the query `a place to cook` locates the kitchen.

Abstract—Robotic tasks such as planning and navigation require a hierarchical semantic understanding of a scene, which could include multiple floors and rooms. Current methods primarily focus on object segmentation for 3D scene understanding. However, such methods struggle to segment out topological regions like “kitchen” in the scene. In this work, we introduce a two-step pipeline to solve this problem. First, we extract a topological map, i.e., floorplan of the indoor scene using a novel multi-channel occupancy representation. Then, we generate CLIP-aligned features and semantic labels for every room instance based on the objects it contains using a self-attention transformer. Our language-topology alignment supports natural language querying, e.g., a “place to cook” locates the “kitchen”. We outperform the current state-of-the-art on room segmentation by $\sim 20\%$ and room classification by $\sim 12\%$. Our detailed qualitative analysis and ablation studies provide insights into the problem of joint structural and semantic 3D scene understanding. Project Page: quest-maps.github.io

I. INTRODUCTION

Understanding both the *structure* and the *semantics* of 3D indoor environments is essential for mapping and navigation in real world [1]–[3]. The structure, i.e., the layout or floorplan, of an indoor environment defines its organisation into different rooms or spaces. Each room has a semantic label associated with it, e.g., kitchen, which can be inferred from the semantic labels of the objects inside it. Thus, we can generate a topological understanding of such 3D scenes using connections *both* within and across the rooms, that is, objects-to-room and room-to-room. Such a topological understanding coupled with a semantic understanding helps

The authors acknowledge the support provided by MeitY, Govt. of India, under the project “Capacity building for human resource development in Unmanned Aircraft System (Drone and related Technology)”.

*Denotes authors with equal contribution

¹Robotics Research Center, IIT-Hyderabad, India

²The University of Adelaide, Australia

plan paths across several rooms, when given a free-form natural language query.

Very few existing works address the twin problem of room segmentation and classification, with most focusing on segmentation alone. Floorplan extraction methods attempting to understand *only the structure* of the scene [4], ignore the rich semantics essential for high-level robotic tasks [5], [6]. On the other hand, dense 3D semantic segmentation methods [7], [8] have focused *solely on object-level* semantic segmentation, without considering the room-level structure of indoor scenes [9].

We propose a first-of-its-kind method that generates a natural language-queryable room-level topological and semantic representation of a 3D indoor scene, by segmenting then understanding. We propose a novel multi-channel occupancy representation of 3D point clouds for room segmentation. These channels help in disambiguating the transition regions (e.g., doors) and rooms. We generate room-level features for each room segment using the CLIP [10] embeddings of objects confined in them. We align the room features with natural language descriptions of the room using a self-attention transformer. As a result, our method supports free-form natural language queries to detect relevant rooms. For example, the query `a place to cook` locates the kitchen. We benchmark our method on synthetic datasets like Structured3D [11] and challenging real-world datasets like Matterport3D [12]. We achieve $\sim 20\%$ and $\sim 12\%$ performance gain over the previous SoTA methods on floorplan extraction [4], [13] and room labeling [14] respectively. Our contributions are:

- 1) a novel pipeline to extract *both the topology* (rooms and transition regions) and *semantic room labels* from 3D indoor scenes;

- 2) a novel *multi-channel occupancy representation* of point clouds that improves room and transition region detection;
- 3) an *attention-based* method to generate *representations for rooms based on the CLIP embeddings of objects in them*, enabling natural language room-level queries;
- 4) an extended version of the Matterport3D dataset [12], with *manual annotations of transition regions* in the scenes.

II. RELATED WORK

A. Floorplan Reconstruction Methods

Floorplan reconstruction organises raw sensor data like point clouds, density maps, or RGB-D images into a structured representation of rooms. Classical methods leverage techniques like plane-fitting [15] or model the problem as an energy minimisation problem [16]. The heuristic-based detection of such methods struggles with noisy and incomplete real world 3D data. Deep learning based methods tackle this concern [17]–[19], where some methods [20], [21] such as HEAT [4] use transformers to generate point, edge, or room level proposals to generate a floorplan. These methods do not predict semantics of the indoor environment (i.e., no room labels like *bedroom*, *kitchen*, etc.) and are thus limited in their downstream utility [2].

B. Semantic Segmentation of Floorplans

This includes assigning a semantic label to every detected room, and transition regions like doorways and windows. FloorNet [19] utilises a CNN-DNN architecture and also identifies salient objects in the scene. However, its object set is very limited. RoomFormer [13], the current state-of-the-art, models the problem as a polygon fitting task by regressing to vertices and edges. Although such a formulation suits synthetic scenes [11] with straight edges and regular angles, it struggles on real-world scenes [12] which invariably contain noise, ill defined edges, and curved walls. These methods also struggle to segment rooms with cluttered objects. Since these methods do not perform dense semantic segmentation, they do not have open vocabularies and cannot be queried in natural language.

C. 3D Semantic Segmentation

3D semantic segmentation methods assign a label to 3D pixels or voxels in the scene [22]. Methods like OpenScene [7], [8] use per-pixel image features extracted from posed RGB-D images of a scene and obtain a point-wise task-agnostic scene representation but struggle in differentiating instances of objects. 3D instance segmentation methods [9], [23], [24] distinguish multiple objects belonging to the same semantic category, predicting individual masks for each instance, but do not build structural associations among objects. Thus, they cannot label entire regions like kitchens, bedrooms etc.

D. 3D Scene Understanding and Scene Graphs

For a robot to understand a scene it must combine place and object labels in a hierarchical manner to create a hybrid map [6]. [25] uses an unsupervised geometry-based approach to segment 3D point clouds into objects and meaningful scene structures but its segmentations lack semantics. [26] can recognize new semantic classes online, but does not leverage rich 3D object geometries and is limited to semantic segmentation in 2D space. To unite structure with semantics [27]–[30] seek to build 3D scene graphs where nodes represent objects and edges encode inter-object relationships. To perform downstream tasks we must be able to query these graphs conveniently. ConceptGraphs and others [31]–[34] use LLMs to accomplish this due to their open-vocabulary and reasoning abilities. However, all these methods struggle to segment rooms from unstructured data, and cannot retrieve all objects from a queried room.

III. METHOD

Our pipeline is illustrated in Fig. 2. Our work focuses on an offline setting. Given a set of posed RGB-D images captured in an indoor scene we construct a point cloud and a 3D semantic object instance map using ConceptGraphs [31]. We use this as our foundation to predict room-level segmentation masks and room labels. We formulate the problem of room segmentation as an instance mask prediction problem, predicting masks for both rooms and transition regions. We perform instance segmentation using a Mask R-CNN architecture, that uses our novel multi-channel occupancy representation to predict room segments. We use a self-attention transformer encoder to produce label-aligned features for each room instance, using the CLIP features [10] of the objects inside it. This allows us to query the scene in natural language at a room topology level.

A. Room and Transition Region Segmentation

1) *Multi-Channel Occupancy Representation*: We propose a novel multi-channel input representation for 3D point clouds. Prior works [4], [13], [18] relied solely on *top-down* density maps as a consolidated 2D representation of the complete 3D point cloud. There are several drawbacks to this representation. Firstly, in sparse reconstruction methods, density of a point cloud of an indoor scene depends highly on the quality of the scanning process; the density in a specific region can vary considerably depending on the trajectory and the amount of time spent by the sensor scanning that specific region. Secondly, under relaxed assumptions about the wall thickness of the environment, it is hard to disambiguate key regions in a top-down density map, e.g. walls vs closed doors. Our key insight is to introduce two additional input channels: two sliced occupancy maps built at floor and ceiling heights, respectively. That is, given a pointcloud $\mathcal{P} = \{p_1, p_2, \dots, p_n\}$ consisting of n unordered 3D points, $p_i \in \mathbb{R}^3, i = 1, \dots, n$ represented using Euclidian co-ordinates, we introduce a 3-channel input representation $\{D, O_{ceiling}, O_{floor}\}$. These are illustrated in Part ① and ② of Fig. 2 (blue for ceiling and green for floor). While the

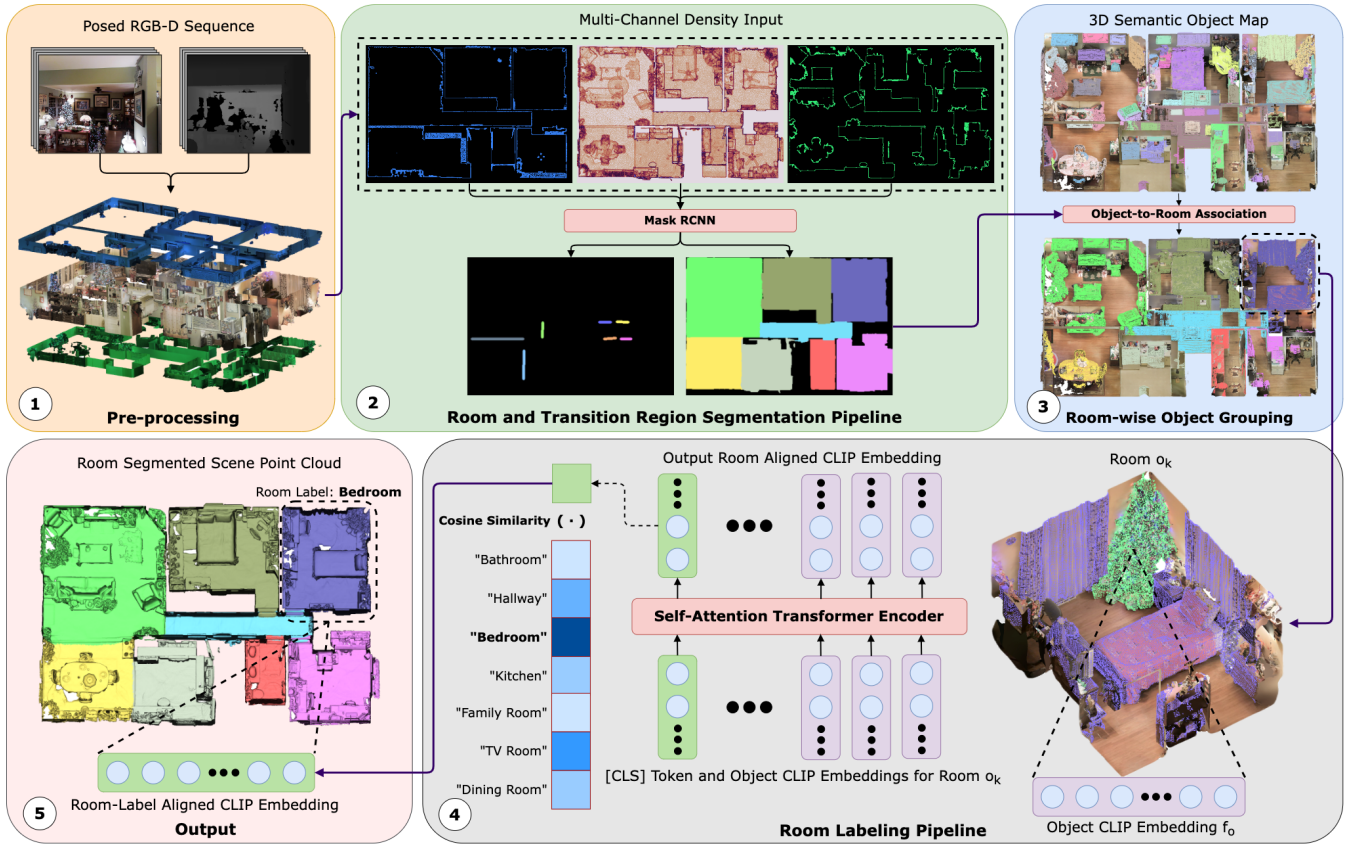


Fig. 2: **Method Overview.** From a 3D scene point cloud reconstructed from a posed RGB-D sequence, our method builds an indoor scene topology map, extracts room and transition region segments, and generates room-label aligned CLIP embeddings and room labels. We propose a novel multi-channel input representation and use an instance segmentation network to predict room masks and detect transition regions. Object level point clouds and CLIP embeddings are generated using an instance mapping pipeline similar to ConceptGraphs [31] and are associated with their rooms from the previous step. A Transformer network takes these object level CLIP embeddings to generate a room label and an aligned CLIP embedding, useful for tasks such as language guided robot navigation with a room-level understanding.

standard technique employed by [13], [18] is used to obtain the top-down density map D , to generate the occupancy slices $\{O_{ceiling}, O_{floor}\}$ we construct a grid of the same resolution as the top-down density map and build occupancy maps of filtered points in a specified range of height values as follows:

$$O_{ij} = \mathbb{I} \left(\sum_{p \in \mathcal{P}} \mathbb{I}(\beta_1 h \leq p_z \leq \beta_2 h) \geq 0 \right) \quad (1)$$

where O_{ij} is the occupancy state of the grid cell, and β_1 and β_2 define the extent of the slice. For the ceiling slice, we choose $\beta_1 = 0.7$ and $\beta_2 = 0.9$, and for the floor slice, we choose $\beta_1 = 0.1$ and $\beta_2 = 0.3$. p represents a point in the point cloud, p_z is the z-coordinate of the point, h is the height of the point cloud calculated as the difference between the ceiling and the floor heights, and $\mathbb{I}(\cdot)$ is the indicator function.

As illustrated in Fig. 2, these additional input channels clearly delineate open doorways as a difference between the floor and ceiling occupancy maps, simplifying the problem of doorway/transition region detection, and consequently room segmentation.

2) *Room-Mask Prediction and Transition Region Detection:* We formulate room instance mask prediction and tran-

sition region detection as a 3-class (2 classes + background) instance segmentation task. The input for this segmentation task is our three-channel 2D array consisting of the density map and the top and bottom occupancy maps, as detailed in Sec. III-A.1. The total loss in Mask R-CNN is given by:

$$\mathcal{L} = \mathcal{L}_c + \mathcal{L}_{BBR} + \mathcal{L}_m \quad (2)$$

where \mathcal{L}_c , \mathcal{L}_{BBR} , and \mathcal{L}_m represent the classification (c), bounding box regression (BBR), and mask (m) losses, respectively.

Classification loss (\mathcal{L}_c) measures the discrepancy between the predicted and ground truth room and transition region class probabilities. *Bounding box regression loss* (\mathcal{L}_{BBR}) penalizes the difference between predicted 2D room and transition region bounding box and ground truth bounding box coordinates of the respective class. *Mask segmentation loss* (\mathcal{L}_m) evaluates the dissimilarity between predicted 2D room/transition masks and their corresponding ground truth binary masks.

B. Room Labeling

We introduce a novel approach to assign room labels to the room masks generated in Sec. III-A. To achieve this, we combine these room segments with their contained objects, where

we obtain object instances using a 3D semantic instance mapping pipeline [31]. Using the semantic information for the objects in a room, we use our model to obtain room embeddings aligned with the CLIP features of room labels.

1) *3D Object Instance Map*: We use the semantic object map creation methodology described in ConceptGraphs [31] to obtain a dense object instance map, as detailed in this sub-section for the sake of completion. This map is derived from a series of posed RGB-D images, collectively represented as $I = \{I_1, I_2, \dots, I_n\}$; $I_t = \langle I_t^{rgb}, I_t^{depth}, \theta_t \rangle$. For some $1 \leq t \leq n$. It lifts the outputs of 2D foundation models into 3D by aggregating observations associated across multiple views. This results in distinct point clouds and features for each object. Segment Anything Model (SAM) [35] is used to obtain class-agnostic instance masks within each image, denoted as $\{m_{t,i}\}_{i=1}^M = \text{Seg}(I_t^{rgb})$. The corresponding mask features $f_{t,i} = \text{Embed}(I_t^{rgb}, m_{t,i})$ are computed using CLIP [10] by cropping the masked image. Each object within the scene, o_j , is thus defined by a tuple $o_j = \langle p_{o_j}, f_{o_j} \rangle$, where p_{o_j} represents the object’s point cloud and f_{o_j} represents the associated CLIP embedding for the object. This point cloud p_{o_j} fuses the constituent point clouds p_i through multi-view projection of its 2D masks $m_{t,i}$ into 3D using depth I_t^{depth} and pose θ_t . In doing so, CLIP features $f_{t,i}$ are averaged across views to obtain a single object CLIP feature vector $f_{o_j} \in \mathbb{R}^D$, $D = 1024$. This feature encapsulates the semantic attributes of the object.

2) *Object-to-Room Association*: After obtaining the room segmentation masks $R = \{r_1, r_2, \dots, r_n\}$ as detailed in Sec. III-A, we allocate object instances to specific rooms. We apply a transformation to the segmentation masks to convert from 2D image space back to 3D world co-ordinates. This transformation is facilitated by the offset and tile-size metadata we save during the point cloud pre-processing step detailed in subsection III-A.1

We employ a KD Tree construction for each room mask, to accurately relate discrete object instances to their corresponding room masks. incorporating all (x, y) coordinates as defined in the world frame. For each object identified within the scene, we compute the centroid of its point cloud. This centroid serves as a query point against the constructed KD Trees, with the closest match dictating the room assignment for the respective object.

3) *Assigning Room Labels*: For each room r_k , we have a subset of objects $\{o_1^{r_k}, o_2^{r_k}, \dots, o_u^{r_k}\}$. The CLIP features for these objects are encoded as a sequence of tokens, serving as input to a self-attending transformer encoder. We prepend a CLS (classification) token for room segment r_k , to the sequence of object feature vectors, represented as $S_k = [CLS; f_{o_1^{r_k}}; f_{o_2^{r_k}}; \dots]$. The purpose of this token is to aggregate information across the sequence, with the attention weights determining the extent of each object’s contribution. The CLS token encapsulates the room’s semantic characteristics as $E_k = T_{enc}(S_k)$ where E_k is the encoded representation.

During the training phase, we compute the cosine similarity between the CLS token $e_{CLS} = E_k[0]$, $e_{CLS} \in \mathbb{R}^D$

and the CLIP text embeddings of various room phrases, like bedroom, $T = \{t_1, t_2, \dots, t_v\}$, $t_v \in \mathbb{R}^D$.

We aim to maximize the similarity between the CLS token embedding and the ground truth room label’s text embedding using a contrastive loss function, similar to the Normalized Temperature-scaled Cross Entropy Loss. This refines the model’s ability to output a room label aligned embedding e_{CLS} , which is in the CLIP embedding space.

$$\mathcal{L}(e_{CLS}, t_{pos}) = -\log \frac{\exp(\text{sim}(e_{CLS}, t_{pos})/\tau)}{\sum_j \exp(\text{sim}(e_{CLS}, t_j)/\tau)} \quad (3)$$

During inference, the room label is determined by identifying the room type phrase embedding that has the highest cosine similarity with the CLS token output embedding. We assign this inferred label to the 3D region.

$$\text{label} = \arg \max_{t \in T} \text{sim}(e_{CLS}, t) \quad (4)$$

Additionally, the output is designated as the room embedding for the associated 3D region. This facilitates a targeted search for specific room types using natural language queries, since the room embedding is aligned with CLIP.

C. Training Details

The Mask R-CNN model used for transition region and room mask segmentation uses a ResNet50 architecture as its backbone, utilizing the layer outputs 1, 2, 3, and 4 of the backbone. The model is configured with an anchor generator for the Region Proposal Network (RPN), a box RoI pooler, and a mask RoI pooler. We initialize the anchor generator with sizes set to (32,), (64,), (128,), and (256,) pixels, and the aspect ratios are configured to 0.5, 1.0, and 2.0 for each size. To reduce detection redundancy, we run Non-Maximum Suppression (NMS) with a threshold of 0.5. AdamW was used as the optimizer with a learning rate $\alpha = 2 \times 10^{-4}$.

Our self-attention transformer model for room labeling incorporates an embedding size of 1024, with 8 attention heads, and 8 layers. During training, we set the dropout at a rate of 0.2 to prevent overfitting and the temperature τ for the loss function to 0.5. Optimization is performed using AdamW, with a learning rate of 1×10^{-5} for 400 epochs.

IV. EXPERIMENTS AND RESULTS

Sec. IV-A discusses datasets, metrics, ablations, results, and comparisons of the room segmentation and transition region detection step. Sec. IV-B discusses labeling rooms and finding room labels for each room mask. In Sec. IV-C we evaluate the complete pipeline and benchmark it. We also show qualitative comparisons to highlight our method’s ability to understand the topology of 3D scenes.

A. Room and Transition Region Segmentation

This part of the topological segmentation pipeline, utilises a MaskRCNN [17] to segment a scene into instance-separated rooms as well as instance-separated doors, i.e. transitions between the rooms.

Datasets. We evaluate our method on two datasets- Structured3D and Matterport3D [11], [12]. **Structured3D**

comprises panoramic Manhattan and non-Manhattan floor-plan layouts with semantically rich room annotations with synthetic scenes with minimal clutter. Notably, the rooms within this dataset are geometrically defined as polygons, a feature that has facilitated the efficacy of previous works [4], [13] based on polygon fitting techniques. We adhere to the official split of 3000 training, 250 validation, and 250 test samples. **Matterport3D** provides RGBD scans and navigable meshes of real-world indoor scenes with object and room type annotations. It is used extensively for robotic tasks like navigation and 3D scene understanding. It offers very detailed scenes with cluttered rooms. The major challenges associated with it are the smaller number of scenes (90), class imbalance in room types, and noisy meshes. Although Matterport3D is a popular dataset for 3D indoor scene understanding, no other work discusses topological segmentation of scenes for this dataset. Since Matterport3D does not provide annotations for transition regions, we manually annotated all doorways in the dataset. We share this extension to the dataset with the community.

Evaluation Metrics. Since we model the room segmentation task as a region-based instance segmentation problem, we use the Average Precision (AP) @ 0.5 IoU threshold as it considers both the accuracy of positive predictions and the ability to capture all instances of a target object. Since the baselines we compare against [4], [13] output an ordered set of vertex coordinates, we consider their predicted mask as a polygon constructed by connecting such vertices. [4] does not segment transition regions, hence the cells have been left blank in Table I.

TABLE I: Quantitative Results for Room Segmentation.

Method	Structured3D		Matterport3D	
	Transitions (AP)	Rooms (AP)	Transitions (AP)	Rooms (AP)
HEAT [4]	-	94.24	-	30.78
RoomFormer [13]	76.94	96.93	24.85	64.42
Ours (Single Channel)	85.60	94.90	50.47	84.83
Ours (Multi Channel)	87.99	98.86	60.92	88.47

Baselines and Results. We primarily compare our model against the state-of-the-art RoomFormer [13] and HEAT [4]. Both are transformer-based methods that fit polygons and output an ordered set of vertices. [4] first detects room corners, and then predicts edges, but no doorways. Results are summarized in Table I, where we outperform the average precision of the state-of-the-art RoomFormer by 1.9% on Structured3D and by 20% on Matterport3D. In addition to rooms, RoomFormer also detects transition regions i.e. doorways. Our method registers a 10% higher average precision than the SoTA RoomFormer on doorway detection. To demonstrate the effectiveness of our multi-channel representation (top and bottom occupancy maps), we construct a baseline that learns only on a single-channel density map. As in Table I, our ablations exhibit an improvement in both doorway detection and room segmentation tasks.

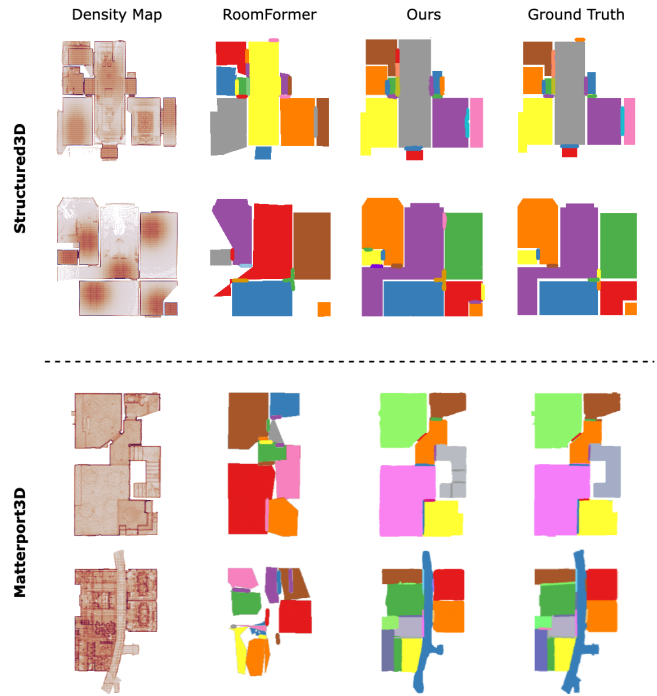


Fig. 3: **Qualitative comparisons on room segmentation task.** Our proposed method is able to handle adverse cases such as regions of low density (introduced by the scanning strategy) in the Structured3D dataset, as well as curved walls and clutter as seen in the Matterport3D dataset. Colors are only representative of the instance separation.

B. Room Labeling

Datasets. In this section we conduct our experiments on the Matterport 3D (MP3D) dataset. Other datasets like Structured3D do not provide posed RGB-D sequences of the scene. So, MP3D serves as the foundation for our model evaluation and facilitates a comprehensive comparison with existing approaches.

Evaluation Metrics. We evaluate performance using two metrics: the weighted average F1 score and Mean Average Precision (mAP), across 23 room label types present in the dataset. Metrics for learning-based approaches are calculated after training the model on the MP3D dataset’s train split (80%), and evaluated on the test split (20%).

Baseline Methods. To benchmark our room labeling model separately, we compare it against different strategies within the domain. As the previous works used ground-truth object-to-room association, all baselines in this study are given the same privilege to ensure a fair comparison.

- 1) *Statistical Baseline:* This model estimates room labels by calculating the product of conditional probabilities for each object present, using ground truth object co-occurrence data [14].
- 2) *Zero-Shot LLM Baseline:* Utilizes [36] to generate descriptive strings about room compositions and applies a scoring method to identify the most appropriate room label [14].
- 3) *Embedding-Based Fine-Tuning:* We use RoBERTa [37] to create embeddings from strings describing objects in a room. These embeddings are then used to predict room types through a multi-layer perceptron [14].

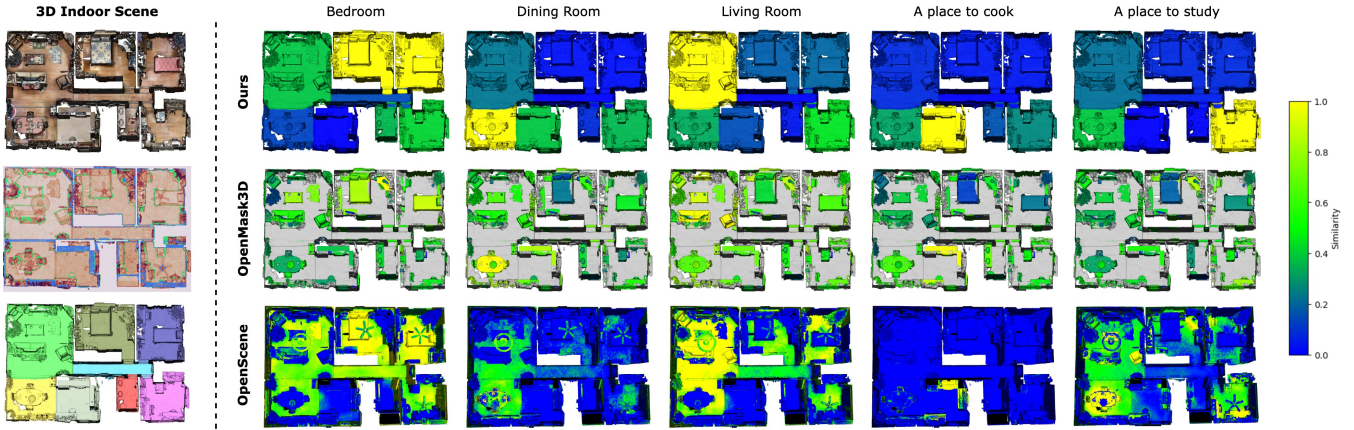


Fig. 4: **Qualitative results on Matterport3D.** Left: Original 3D Scene Point cloud, our multi-channel input representation, and the predicted room segmentation masks for reference. Right: Qualitative comparisons for various room label queries. Here sections of the point cloud are colored based on the similarity of the corresponding query embedding. Color progression from blue to green to yellow signify low to medium to high similarity scores.

- 4) *GNN Baseline*: Objects within a room are encoded as a graph, with nodes connected based on the proximity of objects. The graph is passed through SAGEConv [38] layers to predict room label logits, which are aggregated to determine the final room classification [39].

Ablation Study. To assess the impact of using a self-attention transformer to find the room-label aligned embedding, we do an ablation study, where the room embedding is found by averaging the object CLIP embeddings for all the objects present in that room. In this study, we assign the room label by calculating the cosine similarity of the averaged embedding with the CLIP text embeddings of room phrases. The results are included in Table II in the third last row. Additionally, we test a variant of our method where we add a simple single-layer perceptron to our model’s architecture to predict logits for the room labels directly from the output e_{CLS} token embedding, trained using cross-entropy loss. The results for this are included in Table II in the second last row.

TABLE II: Comparison of Room Labeling Models.

Method	Precision	Recall	F1	mAP
Statistical [14]	49.49	50.50	42.48	28.35
Zero Shot LLM [14]	46.69	45.34	43.00	48.07
Embedding-Based [14]	60.07	66.40	62.75	67.17
GNN Model [39]	58.79	65.59	61.38	67.45
Averaged Object Embeddings	66.28	60.98	60.94	69.02
Ours (Logits)	75.31	75.61	74.63	79.04
Ours (Contrastive)	77.97	76.83	75.43	79.12

Results. As shown in Table II, our proposed CLIP Self-Attention Transformer model outperforms the next best algorithm, the embedding-based fine-tuning approach, by 10-12% in both F1 and mAP metrics. This improvement is significant, especially for room types with fewer examples, addressing the challenge of long-tail distribution. This specifically highlights the utility of our model in finding the room label-aligned embedding.

C. Complete Pipeline

We evaluate our end-to-end pipeline on the Matterport3D dataset. We show the results for room segmentation and

TABLE III: Complete Pipeline Evaluation.

Method	mAP @ 0.5 IoU
RoomFormer [13]	23.52
GT Room Masks + Our Room labeling	79.12
Ours (Complete Pipeline)	74.02

labeling in Table III and compare it against the rich semantic results from RoomFormer [13], the current SoTA in room segmentation and labeling. To demonstrate the difference between the output of our method and other open-vocabulary 3D scene-understanding methods, we show a qualitative comparison in Fig. 4. The comparison with OpenScene [7] shows that as the model computes per-point features, it is unable to correctly highlight the dense 3D region for a specific room type. We also contrast our results with the output from the 3D instance segmentation method OpenMask3D [9], to show that it cannot highlight dense regions given a room type. Although it can correctly identify objects for simpler queries like bedroom and kitchen, it struggles with concepts like living room and study room. These results clearly highlight the difference in the room-level understanding of 3D scenes between previously published methods and our proposed pipeline. Moreover, our room and transition region outputs and room label outputs, enable the downstream applications such as natural-language laden instructions for navigation.

V. LIMITATIONS AND CONCLUSION

In cases where the room was not visible from a top-down view (due to noise or due to the room being very small), the room segmentation pipeline was unable to recognise it. In cases where there were no objects identified in a room by the instance mapping pipeline, the labeling model was unable to generate a CLIP vector for the room. Some environments such as open plan houses are better organised into “zones” rather than “rooms”. Since we rely on walls to segment, we cannot identify sub-room “zones”.

To summarise, our model extracts a natural-language queryable semantic topological map of 3D indoor scenes. We

utilise a 3 channel Mask RCNN-based segmentation model to identify rooms and transition regions. We then utilise a self-attention transformer to generate room-label aligned CLIP features for our rooms based on their contained objects. We outperform the SoTA floorplan segmentation and labeling models on two challenging datasets. Additionally, we enable open-vocabulary room-level natural language queries on the scene. Future work may explore methods to segment a scene into zones. Language-queryable topological segmentation of a 3D scene into rooms can enable a robot to better reason about localization and enhance downstream tasks like planning [40] and navigation [32]; integration of our method in such pipelines will be an exciting future work.

REFERENCES

- [1] X. Han, S. Li, X. Wang, and W. Zhou, "Semantic mapping for mobile robots in indoor scenes: A survey," *Information*, vol. 12, 2021.
- [2] J. Crespo, J. C. Castillo, O. M. Mozos, and R. Barber, "Semantic information for robot navigation: A survey," *Applied Sciences*, vol. 10, no. 2, p. 497, 2020.
- [3] N. Zimmerman, T. Guadagnino, X. Chen, J. Behley, and C. Stachniss, "Long-term localization using semantic cues in floor plan maps," *IEEE Robotics and Automation Letters*, vol. 8, no. 1, pp. 176–183, 2022.
- [4] J. Chen, Y. Qian, and Y. Furukawa, "Heat: Holistic edge attention transformer for structured reconstruction," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [5] K. Muravyev and K. Yakovlev, "Evaluation of topological mapping methods in indoor environments," *IEEE Access*, vol. 11, pp. 132 683–132 698, 2023.
- [6] S. Garg, N. Sünderhauf, F. Dayoub, D. Morrison, A. Cosgun, G. Carneiro, Q. Wu, T.-J. Chin, I. Reid, S. Gould *et al.*, "Semantics for robotic mapping, perception and interaction: A survey," *Foundations and Trends® in Robotics*, vol. 8, no. 1–2, pp. 1–224, 2020.
- [7] S. Peng, K. Genova, C. Jiang, A. Tagliasacchi, M. Pollefeys, T. Funkhouser *et al.*, "Openscene: 3d scene understanding with open vocabularies," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 815–824.
- [8] D. Seichter, M. Köhler, B. Lewandowski, T. Wengefeld, and H.-M. Gross, "Efficient rgb-d semantic segmentation for indoor scene analysis," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 13 525–13 531.
- [9] A. Takmaz, E. Fedele, R. Sumner, M. Pollefeys, F. Tombari, and F. Engelmann, "Openmask3d: Open-vocabulary 3d instance segmentation," *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [10] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.
- [11] J. Zheng, J. Zhang, J. Li, R. Tang, S. Gao, and Z. Zhou, "Structured3d: A large photo-realistic dataset for structured 3d modeling," in *Computer Vision—ECCV 2020*, 2020, pp. 519–535.
- [12] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," *3DV*, 2017.
- [13] Y. Yue, T. Kontogianni, K. Schindler, and F. Engelmann, "Connecting the Dots: Floorplan Reconstruction Using Two-Level Queries," in *CVPR*, 2023.
- [14] W. Chen, S. Hu, R. Talak, and L. Carlone, "Leveraging large (visual) language models for robot 3d scene understanding," 2023.
- [15] C. Mura, O. Mattausch, A. J. Villanueva, E. Gobbetti, and R. Pajarola, "Automatic room detection and reconstruction in cluttered indoor environments with complex room layouts," *Computers & Graphics*, vol. 44, pp. 20–32, 2014.
- [16] R. Ambruş, S. Claiici, and A. Wendt, "Automatic room segmentation from unstructured 3-d data of indoor environments," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 749–756, 2017.
- [17] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask r-cnn," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988, 2017.
- [18] J. Chen, C. Liu, J. Wu, and Y. Furukawa, "Floor-sp: Inverse cad for floorplans by sequential room-wise shortest path," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2661–2670.
- [19] C. Liu, J. Wu, and Y. Furukawa, "Floornet: A unified framework for floorplan reconstruction from 3d scans," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 201–217.
- [20] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *ECCV*, 2020.
- [21] J.-W. Su, K.-Y. Tung, C.-H. Peng, P. Wonka, and H.-K. J. Chu, "Slibonet: Floorplan reconstruction via slicing box representation with local geometry regularization," in *Advances in Neural Information Processing Systems*, A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023, pp. 48 781–48 792.
- [22] C. Couprie, C. Farabet, L. Najman, and Y. Lecun, "Indoor Semantic Segmentation using depth information," in *First International Conference on Learning Representations (ICLR 2013)*, 2013.
- [23] L. Han, T. Zheng, L. Xu, and L. Fang, "Occuseg: Occupancy-aware 3d instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2940–2949.
- [24] C. Xie, Y. Xiang, A. Mousavian, and D. Fox, "Unseen object instance segmentation for robotic environments," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1343–1359, 2021.
- [25] T. T. Pham, M. Eich, I. Reid, and G. Wyeth, "Geometrically consistent plane extraction for dense indoor 3d maps segmentation," in *IEEE/RSJ IROS*. IEEE, 2016, pp. 4199–4204.
- [26] N. Sünderhauf, F. Dayoub, S. McMahon, B. Talbot, R. Schulz, P. Corke, G. Wyeth, B. Upcroft, and M. Milford, "Place categorization and semantic mapping on a mobile robot," in *ICRA*. IEEE, 2016.
- [27] T. T. Pham, T.-T. Do, N. Sünderhauf, and I. Reid, "Scenecut: Joint geometric and object segmentation for indoor scenes," in *ICRA*. IEEE, 2018, pp. 3213–3220.
- [28] H. Bavlle, J. L. Sanchez-Lopez, M. Shaheer, J. Civera, and H. Voos, "Situational graphs for robot navigation in structured indoor environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, 2022.
- [29] 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/CVF International Conference on Computer Vision (ICCV)*, October 2019.
- [30] Z. Liao, Y. Zhang, J. Luo, and W. Yuan, "Tsm: Topological scene map for representation in indoor environment understanding," *IEEE Access*, vol. 8, pp. 185 870–185 884, 2020.
- [31] Q. Gu, A. Kuwajerwala, S. Morin, K. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa, C. Gan, C. de Melo, J. Tenenbaum, A. Torralba, F. Shkurti, and L. Paull, "Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning," in *ICRA*, 2024.
- [32] S. Garg, K. Rana, M. Hosseinzadeh, L. Mares, N. Sünderhauf, F. Dayoub, and I. Reid, "Robohop: Segment-based topological map representation for open-world visual navigation," in *ICRA*, 2024.
- [33] B. Chen, F. Xia, B. Ichter, K. Rao, K. Gopalakrishnan, M. S. Ryoo, A. Stone, and D. Kappler, "Open-vocabulary queryable scene representations for real world planning," in *ICRA*. IEEE, 2023.
- [34] J. Strader, N. Hughes, W. Chen, A. Speranzon, and L. Carlone, "Indoor and outdoor 3d scene graph generation via language-enabled spatial ontologies," 2023.
- [35] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," in *ICCV*, 2023, pp. 4015–4026.
- [36] S. Black, L. Gao, P. Wang, and S. Biderman, "Gpt-neo: Large scale autoregressive language modeling with mesh-tensorflow," 2021.
- [37] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," 2019, cite arxiv:1907.11692.
- [38] W. L. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *NeurIPS*, 2017.
- [39] G. Kumar, N. S. Shankar, H. Didwania, R. Roychoudhury, B. Bhowmick, and K. M. Krishna, "Gcexp: Goal-conditioned exploration for object goal navigation," in *2021 30th IEEE RO-MAN*, 2021.
- [40] K. Rana, J. Haviland, S. Garg, J. Abou-Chakra, I. Reid, and N. Sünderhauf, "Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning," in *7th Annual Conference on Robot Learning*, 2023.