External Knowledge Enhanced 3D Scene Generation from Sketch (Supplementary Material)

Zijie Wu[1*], Mingtao Feng[2], Yaonan Wang[1], He Xie[1], Weisheng Dong[2], Bo Miao[3], and Ajmal Mian[3]

[1] Hunan University
[2] Xidian University
[3] University of Western Australia

Here, we provide more details about our proposed method, covering both explicit and implicit scene generation (Sec. A) from freehand sketch and entities. Additionally, we offer more details of our proposed knowledge base in Section B. In Section C, we present an inconsistency test between sketch and entity selection. We provide two visualization examples illustrating missing furniture in the sketch and missing furniture in the entity. Moreover, we introduce a solution for the concatenation of condition designs to equip the model with the ability for self-correction (Sec. D). It is important to note that all generations discussed here are *fully generated*. Finally, we give more details of our evaluation metrics and conduct an additional comparison using human evaluation as metric in Sec. E.

A Sketch based Knowledge enabled Scene Generation

Returning to the scene generation ability of our proposed model, in this section, we present additional visualizations for 3D generation from freehand sketch and selected entities in scene generation (shown in Figure A), as well as for scene completion (both explicit completion and implicit generation shown in Figure B). *We ensure that the edge detection sketches match the hand drawn ones.* Our conditional method produces plausible 3D scenes that are more faithful to the input sketch descriptions. Note that all presented scenes are generated by our model in an end-to-end manner without any retrieval, resulting in realistic and high-quality scenes.

B Details of Knowledge Base

Here, we give the construction details of knowledge base, including entity candidates for different scene categories and the knowledge base specifics.

B.1 Scene Categories

We construct the knowledge base from two datasets in our paper. The object types from which the knowledge base is constructed are listed below:

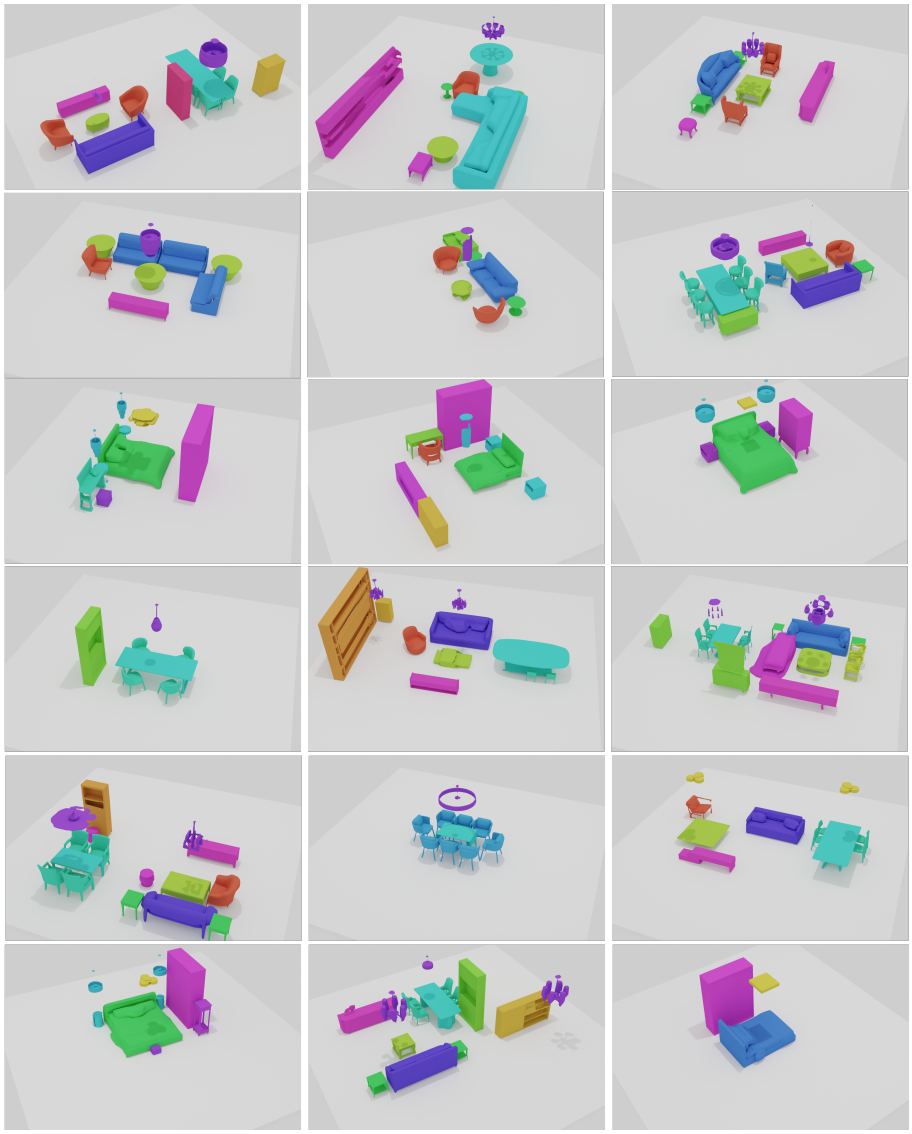

Fig. A: Additional scene generation results from input sketch (not shown here).

 # 3D-FRONT-Categories:

[‘armchair’, ‘bookshelf’, ‘cabinet’, ‘ceiling lamp’, ‘chair’, ‘chaise longue sofa’, ‘children cabinet’, ‘chinese chair’, ‘coffee table’, ‘console table’, ‘corner side table’, ‘desk’, ‘dining chair’, ‘dining table’, ‘double bed’, ‘dressing chair’, ‘dressing table’, ‘kids bed’, ‘l-shaped sofa’, ‘lazy sofa’, ‘lounge chair’, ‘loveseat sofa’, ‘multi seat sofa’, ‘nightstand’, ‘pendant lamp’, ‘round end table’, ‘shelf’, ‘single bed’, ‘sofa’, ‘stool’, ‘table’, ‘tv stand’, ‘wardrobe’, ‘wine cabinet’]

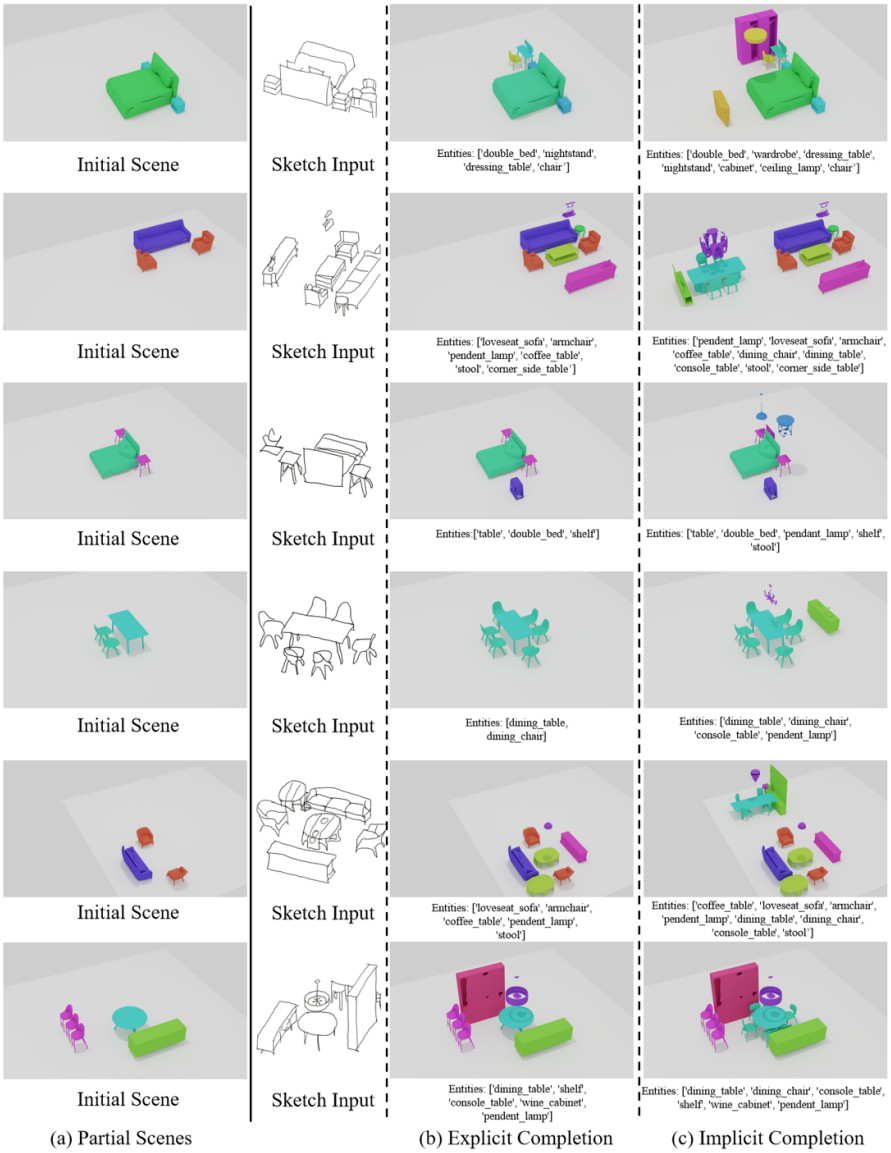

Fig. B: Additional scene completion results. Input partial scenes (a) are completed based on sketches (2nd col). In (b), a scene is explicitly completed by defining additional objects in the sketch and mentioning them as entities. In (c), a scene is implicitly completed by mentioning entities that are not visible in the sketch.

 # ScanNet-Categories:

['chair', 'table', 'cabinet', 'desk', 'sofa', 'lamp', 'bed', 'bookshelf']

In our knowledge transfer experiment, we merge the similar nodes in knowledge base when extracting knowledge on 3DFRONT for generating ScanNet scenes.

Scene Clustering: We construct the knowledge base with five predefined relationships. For each scene in dataset, DBSCAN divides the scene into different clusters, which is shown in C. Subsequently, we extract the knowledge relationships for both 'adjacent' and 'distant' to build the knowledge base upon the clustered object groups.

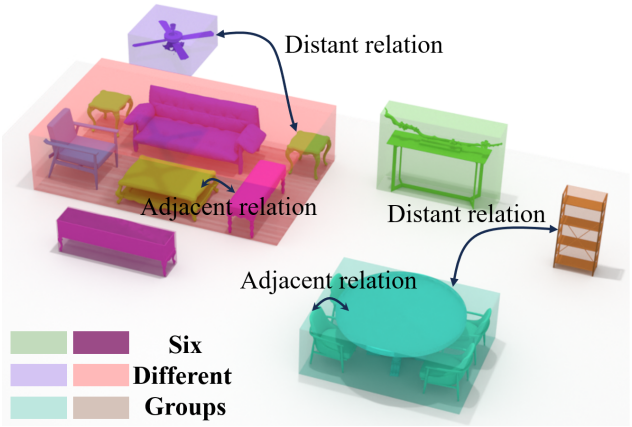

Fig. C: Visualization of how the objects are clustered into 6 groups. We also show only two relationships here i.e. distant and adjacent.

C Inconsistency Test Between Sketch & Entity Selection

We present an inconsistency test between the sketch and entity selection. Initially, we provide an incomplete hand-drawn sketch compared to the selected entities, illustrated in Fig. D. This can be considered a form of *implicit completion* in the scene completion task.

Additionally, we visualize situations where the selected entities are insufficient for sketch-based scene reasoning, as demonstrated in Fig. E. Three types of generations occur when there is a missing number of entities for knowledge reasoning:

1) When the hand-drawn sketch provides enough information, a reasonable scene is generated.

2) If the hand-drawn sketch lacks sufficient information, the furniture corresponding to the missing entities will not be generated.

3) Through a combination of existing knowledge, our model produces furniture similar to the drawn items. However, missing knowledge for entities can result in the generation of unreasonable relationships (Fig. E(d) where the drawn desk is generated as a coffee table, creating a crowded layout).

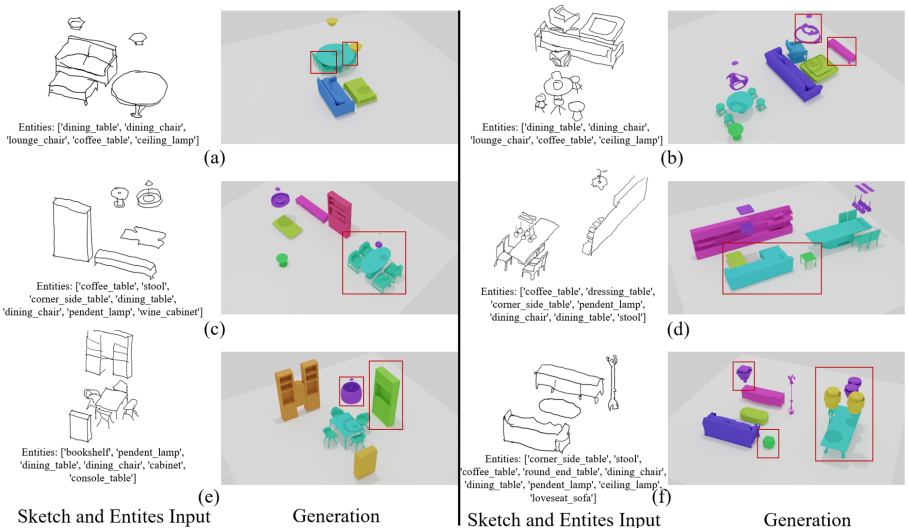

Fig. D: Results for miss aligned inputs - additional entities. The hand drawn sketch is incomplete and does not contain all the entities e.g. missing dining-chair in the sketch in (a). Our model still generates a plausible 3D scene that is aligned with the sketch and contains the *invisible* objects.

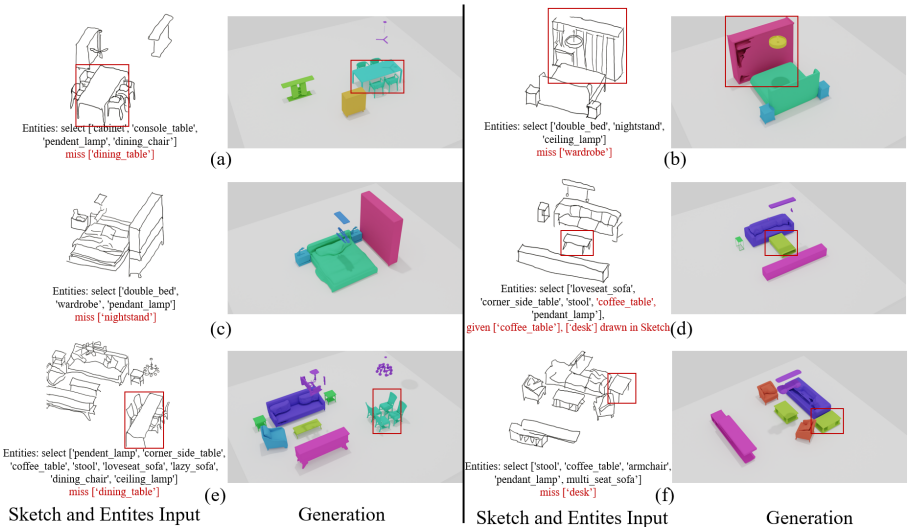

Fig. E: Results for missed aligned inputs - missing entities. The hand drawn sketch contains entities that are mentioned e.g. in (a) the dining table is present in the sketch but not mentioned as an entity. Additionally, some items may not match e.g. in (d) the sketch contains a desk, whereas the entity is 'coffeetable'.

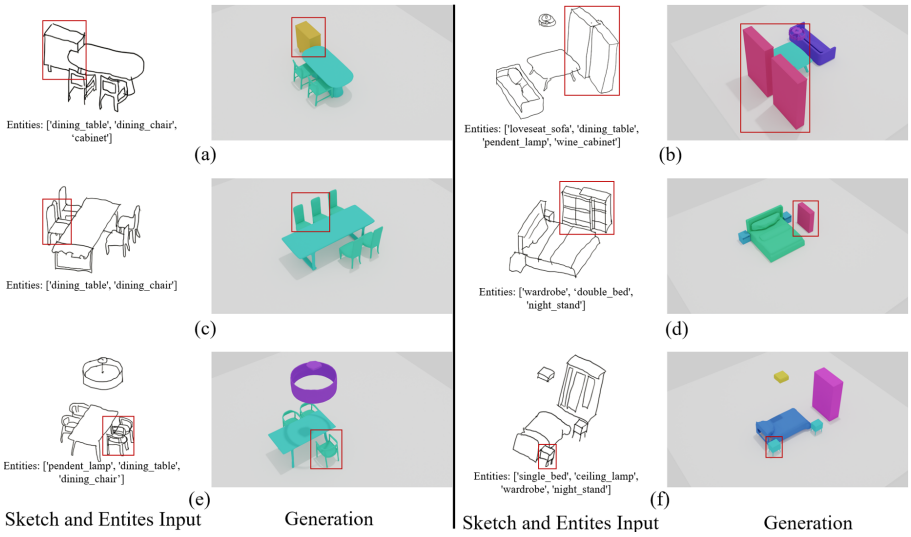

Fig. F: Self-correction for hand-drawn mistakes in sketch.

D Self-correctional Condition Design

As our proposed model in the paper strives for faithful generation to a sketch, we concatenate the sketch feature and graph feature in the embedding context $\mathcal{I} = [c, t, \mathcal{O}t] \in \mathbb{R}^{D\times(I)}$, where $I = M + 3$ in the paper and $I = M + 2$ here. The c_{sketch} and c_{graph} are attended to in the transformer framework. However, this is under the strong assumption that users never make mistakes. The overlapping furniture in the sketch results in unrealistic generation reflected in the output scenes. Hence, we propose a new conditional design to present the interesting feature that allows the model to self-correct the mistakes in the user instructions:

$$c = [H^S \in \mathbb{R}^{1/2D\times1}; H^G \in \mathbb{R}^{1/2D\times1}] \in \mathbb{R}^{D\times1}, \tag{1}$$

The self-correctional condition design disrupts the attention mechanism between the sketch condition and graph condition. It produces more reasonable and realistic scenes, even when the hand-drawn sketch contains errors. We visualize the self-correctional generation in Fig F. The overlapped hand-drawn sketches introduce errors when exposing the spatial location of furniture. The model in the paper would faithfully follow the instructions even if the furniture overlaps in the sketch, leading to unreasonable generation. With the self-correctional condition design, when presented with erroneous sketches, the model refers to the learned scenes and knowledge to replace the erroneous furniture. As shown in Fig F (a-c,f), the two pieces of furniture are replaced with appropriate spacing instead of being *overlapped* in the sketch. In Fig F (d,e), errors for the same categories, such as two *overlapped wardrobes* and two *overlapped dining chairs* in the sketch, are corrected as a single piece of furniture that is reasonable relative to its neighbors.

E Evaluation Metrics

Following previous works [2] [3] [1], we use Frechet Inception Distance (FID), Kernel Inception Distance (KID $\times$ 0.001), Scene Classification Accuracy (SCA), and Category KL Divergence (CKL $\times$ 0.01) to measure the plausibility and diversity of 1,000 generated scenes. For FID, KID, and SCA, we render the generated and ground-truth scenes into 256^2 semantic maps using top-down orthographic projections, where the texture of each object is uniquely determined by the associated color of its semantic class. We use a uniform camera/rendering setting for all methods to ensure a fair comparison. For CKL, we calculate the KL divergence between the semantic class distributions of synthesized and ground-truth scenes. For FID, KID, and CKL, lower numbers denote better approximation of the data distribution. FID and KID can also indicate diversity. For SCA, a score close to 50% represents that the generated scene is indistinguishable from real scenes.

Furthermore, We also conducted human evaluations for comparison with DiffuScene, graph-to-3D, and ATISS. Table A shows the results, where 30 people rated the generations on a scale of 1 to 5 (higher is better). SEK outperforms the baseline methods in all individual scene categories and achieves an average rating of 4.3, compared to the average rating of 3.36 by the nearest competitor, DiffuScene. Our method provides more satisfactory results for users.

Table A: Human evaluations: 100 generated objects rated on a scale of 1 to 5 by 30 people.

Human rate (↑)	graph-to-3D	ATISS	DiffuScene	SEK
Bedroom	2.0	3.0	3.2	**4.4**
Dining Room	2.4	3.5	3.6	**4.0**
Living Room	3.1	2.4	3.3	**4.5**

References

1. Paschalidou, D., Kar, A., Shugrina, M., Kreis, K., Geiger, A., Fidler, S.: Atiss: Autoregressive transformers for indoor scene synthesis. Advances in Neural Information Processing Systems **34**, 12013–12026 (2021) 7
2. Wang, X., Yeshwanth, C., Nießner, M.: Sceneformer: Indoor scene generation with transformers. In: 2021 International Conference on 3D Vision (3DV). pp. 106–115. IEEE (2021) 7
3. Yang, H., Zhang, Z., Yan, S., Huang, H., Ma, C., Zheng, Y., Bajaj, C., Huang, Q.: Scene synthesis via uncertainty-driven attribute synchronization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5630–5640 (2021) 7