External Knowledge Enhanced 3D Scene Generation from Sketch

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

Abstract. Generating realistic 3D scenes is challenging due to the complexity of room layouts and object geometries. We propose a sketch based knowledge enhanced diffusion architecture (SEK) for generating customized, diverse, and plausible 3D scenes. SEK conditions the denoising process with a hand-drawn sketch of the target scene and cues from an object relationship knowledge base. We first construct an external knowledge base containing object relationships and then leverage knowledge enhanced graph reasoning to assist our model in understanding hand-drawn sketches. A scene is represented as a combination of 3D objects and their relationships, and then incrementally diffused to reach a Gaussian distribution. We propose a 3D denoising scene transformer that learns to reverse the diffusion process, conditioned by a hand-drawn sketch along with knowledge cues, to regressively generate the scene including the 3D object instances as well as their layout. Experiments on the 3D-FRONT dataset show that our model improves FID, CKL by 17.41%, 37.18% in 3D scene generation and FID, KID by 19.12%, 20.06% in 3D scene completion compared to the nearest competitor DiffuScene.

Keywords: Scene Generation · Knowledge Enhanced System · Diffusion

1 Introduction

There is an increasing demand for tools that automate the creation of artificial 3D environments for applications in game development, movies, augmented/virtual reality, and interior design. Sketch based 3D scene generation allows users to control the generated scene entities through a rough hand-drawn sketch. Several methods for 3D scene generation rely on an input image [34, 40] to guide the generation process for alignment with the input. However, such methods focus on leveraging 2D-3D consistency for supervision, which restricts diversity in the generated scene. Moreover, obtaining an image that serves as a 2D rendering of the intended 3D scene is not always straightforward.

Recently, sketch based methods [32, 43, 51, 59] have been proposed for user-specified 3D modeling. However, these methods focus primarily on generating

* Work performed during visit at the University of Western Australia
✉ Corresponding author.

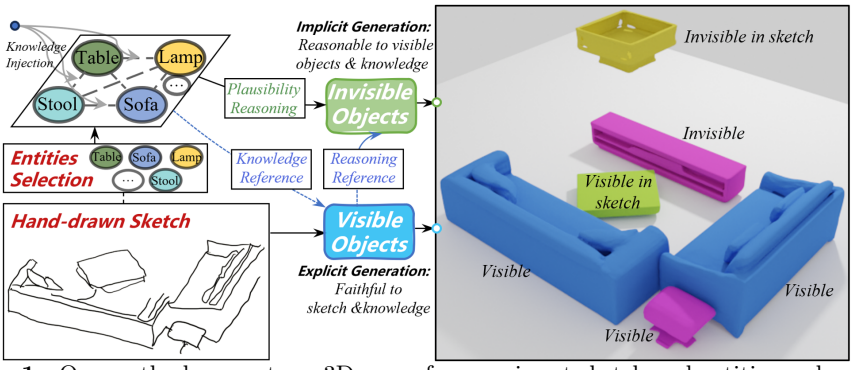

Fig. 1: Our method generates a 3D scene from an input sketch and entities, enhanced by external knowledge. It follows explicit visual cues in the sketch for *visible* objects along with their relationships and employs plausibility reasoning to add objects that are not explicitly depicted (*invisible*) in the sketch, to generate a coherent scene.

single 3D objects. While much progress has been made in generating high-quality 3D objects, generation of complete 3D scenes is still challenging given the complex scene layouts, diverse object geometries and strong coherence between objects. For example, a chair can be placed underneath a table or around it, or may even be to the side of a bed. Various arrangements are possible for a chair and its neighboring objects, and each one must follow some rules including object-object relationships and space occupancy. To improve the representation and comprehension of 3D scenes, external knowledge has been introduced for multiple primary tasks, e.g., scene graph generation [11], robotic grounding [13], visual question answering [36], and semantic segmentation [18]. This involves reusing ontologies and integrating existing knowledge for improved outcomes. External knowledge has been a prevailing technique in transferring implicit representations between scenes for improved performance in various 3D vision tasks. In this paper, we leverage external knowledge to provide auxiliary information for completing implicit scene patterns, that are not obvious in the sparse ambiguous hand-drawn sketch, and guide our proposed indoor 3D scene generation model.

Existing methods use simple hand-crafted object relationships for generating 3D scenes. For instance, GRAINS [25] organize the scene objects into simple scene graph hierarchies that are manually defined. Furthermore, numerous works generate indoor scene layouts [29, 38, 40] in the form of object identities and bounding boxes and then *retrieve* existing furniture shapes from a repository for placement inside those bounding boxes. Hence, the generated layouts as well as the object shapes both lack diversity.

We propose a 3D scene generation method (see Fig. 1) that creates custom, diverse and plausible 3D scenes from hand-drawn sketches and entities, enhanced by external knowledge of object relationships. Our method takes a sketch as the main scene description and leverages external knowledge cues to reduce ambiguity in inferring *visible* objects (shapes and layout) in the sketch and enhance the generation diversity by including *invisible* objects that are not drawn in the sketch. We build an external knowledge base to contain rich knowledge priors

of relationships. *Invisible* objects are inferred from the knowledge base across the *invisible* and *visible* objects to maintain diversity, plausibility and alignment with user specifications. Based on the sketch and knowledge reasoning, the proposed conditional scene diffusion *simultaneously* generates a 3D scene layout with detailed object geometries (see Fig. 2) with plausible structure and coherence among objects. Our contributions are summarized below:

- We propose an end-to-end generative model (SEK) to *simultaneously* generate realistic 3D room layouts and object shapes based on hand-drawn sketches and object entities, enhanced by external knowledge.
- We construct an external knowledge base that defines various object relationships, and serves as a foundational entity-relationship prior to provide additional guidance to the inference process. This improves the plausibility of the generated scenes, including layout and object shapes.
- We learn novel reasoning from external knowledge cues and hand-drawn sketches to extract a relationship subgraph of the specified entities during inference and integrate it with sketch features to form the diffusion condition.
- We propose a 3D denoising scene transformer that operates in the latent space and converts the denoising scene representation into the frequency domain to alleviate the influence of constants corresponding to invalid objects (padded zeros in scene representation) that are added to make the number of objects per scene constant.

2 Related Works

Sketch based 3D Object Generation: Sketches have been used as a sparse representation of natural images and 3D shapes [44, 59] as they are quite illustrative, despite their simplicity and abstract nature. Some works [28] estimate depth and 3D normals from a set of viewpoints for an input sketch, which are then integrated to form a 3D point cloud. Others [20] represent the 3D shape and its occluding contours in a joint VAE latent space during training, enabling them to retrieve a sketch during inference to generate a 3D shape. Recently, Kong *et al.* [23] trained a diffusion model conditioned on sketches using multi-stage training and fine-tuning. Sanghi *et al.* [43] used local semantic features from a frozen large pre-trained image encoder, such as CLIP, to map the sketch into a latent space for diverse shape generation. These methods mainly focus on object level generation, which prefers simple shape information instead of hierarchical relationship generation in scenes. However, 3D scenes contain rich information, including different furniture types, object geometries, room layouts, etc, presenting significant challenges to the generation process.

Knowledge Graphs in 3D scenes: Prior knowledge has proven to be an effective source of information to enhance object and relation recognition [10]. Pioneering works, including ConceptNet [45], VisualGenome [24], DBPedia [2], and WordNet [33], have extensively studied the acquisition of label-pairs frequency as a primary source of relations. These methods have achieved great success in many applications such as image generation [21, 54], visual question answering [8,47], camera localization [1], and robotic grounding [13,19,42]. Nevertheless,

the integrated knowledge is not very useful in isolation since it is hard-coded in the form of intrinsic parameters. Hence, some methods bring external knowledge bases into 3D tasks related inference. Gu *et al.* [15] extracted knowledge triplets from the ConceptNet knowledge bases to help scene graph generation. GBNet [58] adopted auxiliary edges as bridges that facilitates message passing between knowledge graph and scene graph. Li *et al.* [25] introduced hand-crafted relationships for 3D scene generation in a recursive manner. The complete scene is encoded as multiple properties, including geometry and relationship, and is then recovered in a suggested pattern. Although previous studies have taken notice of knowledge in the 3D area, they only implicitly mine the extra knowledge base or define the relationship pairs to strengthen the iterative scene recovery between relationships and objects while ignoring the intrinsic properties of the data for specific 3D scene knowledge representation.

3D Scene Generation: Early methods for 3D scene generation are based on GANs [26, 55], VAEs [5, 26, 39], and Autoregressive models [14, 38, 50]. They are renowned for their ability to generate high-quality results quickly, yet they often face challenges of limited diversity and difficulties in producing samples that align well with user specifications. Numerous methods learn to produce faithful results under different input conditions, such as images [34, 35, 48], text [30, 46], sketches [52], and wall layouts [14, 22, 38]. Another approach in 3D scene generation is based on graph conditioning [7, 29]. Graph-to-3D [7] jointly optimizes models to learn both scene layouts and shapes conditioned on a scene graph. However, the scene graph does not directly reveal the relationships among objects. This necessitates a complete graph description, impacting their realism and applicability. Conditional 3D scene synthesis methods offer faithful scene recovery tailored to user specifications, while generative methods strike a balance between diversity and alignment with user specifications.

3 Diffusion Model for Scene Generation

In the diffusion process, the data distribution is gradually destroyed into Gaussian noise following the Markov forward chain. A denoising process then recovers data from the Gaussian distribution with an iterative reverse chain. To devise our scene diffusion model for generating 3D scenes, we introduce matrix conversion to represent an indoor scene in the form of a matrix. All processes operate on the matrix field. Fig. 2 illustrates how the diffusion and denoising processes mutually transform the Gaussian and target data distributions.

Matrix conversion is proposed to encode the scene objects into parameters that specify their locations and shape attributes. Given an indoor scene $\mathcal{O}$ containing a set of objects $\{\mathbf{o}_i\}|_{i=1}^{K}$, each object o_i is characterized by a 1-D vector which concatenates its spatial location and latent shape attributes, *i.e.* $\mathbf{o}_i = [\mathcal{G}_i, \mathcal{F}_i] \in \mathbb{R}^{D\times 1}$. Every 3D scene is normalized by relocating it in a world coordinate system where the floor center is the origin.

The placement location of each object $\mathcal{G}_i = [\alpha_i, \mathbf{s}_i, \mathbf{t}_i]$ is defined by its axis-aligned 3D bounding box size $\mathbf{s} \in \mathbb{R}^{3\times 1}$, translation $\mathbf{t} \in \mathbb{R}^{3\times 1}$ and yaw angle $\alpha \in \mathbb{R}^{2\times 1}$. Following [57], the yaw angle is parameterized as a 2D vector of sine

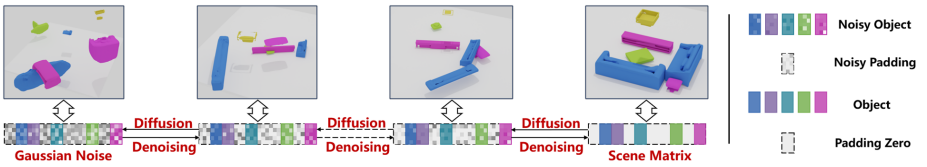

Fig. 2: Demonstration of the scene diffusion/denoising process of the matrix field and the spatial field. The denoising process samples from a Gaussian distribution and progressively denoises the sample for plausible and realistic scene generation. Note how the layout and 3D shapes are both *simultaneously* denoised.

and cosine values. All objects are normalized to $[-1, 1]^3$ first, and subsequently encoded into latent space. The shape latent $\mathcal{F} \in \mathbb{R}^{\mathcal{F}}$ is trained on DeepSDF [37] to obtain a unique code per object.

Since the number of objects can vary across scenes, we pad zero-vectors $\{\mathbf{z}_i | i \in (K+1, M)\} \in \mathbb{R}^{D\times 1}$ so that all scenes have a fixed number of M objects. Here, $K < M$. All objects are concatenated to form a full scene representation: $\mathcal{O} \in \mathbb{R}^{D\times M}$. A scene is encoded as a unique matrix, where each row corresponds to an object with shape, location and size attributes. By synthesizing various combinations of object parameters, we can generate diverse scenes.

Diffusion Process: In the forward chain of scene diffusion, the original scene matrix $\mathcal{O}_0 \sim q(\mathcal{O}_0)$ is gradually corrupted into a pre-defined T-step noised scene distribution following the Markov chain assumption until the Gaussian distribution is reached. Based on the Markov property, the joint distribution $\mathcal{O}_{1:T}$ is straight derived from the original scene matrix $\mathcal{O}_0$:

$$q\left(\mathcal{O}_{0:T}\right) = q\left(\mathcal{O}_0\right) \prod_{t=1}^{T} q\left(\mathcal{O}_t \mid \mathcal{O}_{t-1}\right), \quad q\left(\mathcal{O}_t \mid \mathcal{O}_{t-1}\right) = \mathcal{N}\left(\sqrt{1-\beta_t}\mathcal{O}_{t-1},\ \beta_t \mathbf{I}\right), \quad (1)$$

where $\mathcal{N}\left(\mu, \sigma^2\right)$ denotes a Gaussian distribution and β_t is the known variance defined during the diffusion process.

Deniosing process: Since the forward chain concludes with a Gaussian distribution, we apply the reverse chain starting from a standard Gaussian prior and ending with the desired scene representation $\mathcal{O}_0$:

$$p_\theta\left(\mathcal{O}_{0:T}\right) = p\left(\mathcal{O}_T\right) \prod_{t=1}^{T} p_\theta\left(\mathcal{O}_{t-1} \mid \mathcal{O}_t\right), \quad p_\theta\left(\mathcal{O}_{t-1} \mid \mathcal{O}_t\right) = \mathcal{N}\left(\mu_\theta\left(\mathcal{O}_t, t\right), \sigma_t^2 \mathbf{I}\right), \quad (2)$$

where p_θ is the inference step using parameterized network of the proposed scene denoiser. SEK is trained by minimizing the cross-entropy loss between two diffusion chains in relation to the sketch and knowledge enhanced conditional feature c, and by learning the scene denoiser parameters θ:

$$\min_\theta \mathbb{E}_{\mathcal{O}_0 \sim q(\mathcal{O}_0), \mathcal{O}_{1:T} \sim q(\mathcal{O}_{1:T})} [\sum_{t=1}^{T} \log p_\theta(\mathcal{O}_{t-1} | \mathcal{O}_t, c)]. \quad (3)$$

Following [17] to minimize Eq. 3, SEK learns to match each $q\left(\mathcal{O}_{t-1} \mid \mathcal{O}_t, \mathcal{O}_0\right)$ and $p_\theta\left(\mathcal{O}_{t-1} \mid \mathcal{O}_t\right)$ by estimating the noise ϵ_θ under the condition $(c, t, \mathcal{O}_t)$ to match the added noise ϵ in the diffusion process:

$$\mathcal{L}_{\text{sce}} = \mathbb{E}_{c,t,\epsilon,\mathcal{O}_0}\left[\|\epsilon - \epsilon_\theta\left(c, t, \mathcal{O}_t\right)\|^2\right], \epsilon \sim \mathcal{N}(0, \mathbf{I}). \quad (4)$$

Scene diffusion progressively generates the 3D scene using the reverse chain:

$$\mathcal{O}_{t-1} = 1/\sqrt{\alpha_t}\left(\mathcal{O}_t - 1 - \alpha_t/\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(c,t,\mathcal{O}_t)\right) + \sqrt{\beta_t}\epsilon, \tag{5}$$

where $\alpha_t = 1 - \beta_t$, $\tilde{\alpha} = \prod_{s=1}^{t} \alpha_s$, and ϵ is the standard Gaussian noise. Building upon a well-defined scene diffusion, the proposed model is theoretically capable of generating high-quality and diverse 3D scenes.

4 Knowledge Enhanced Sketch based Guidance

Equipped with the scene diffusion model, the problem we need to address is how to ensure that the generated scene aligns with user description while preserving diversity, quality and plausibility. The input modality requires capturing the essence of the scene, providing a comprehensive description of the backbone while allowing flexibility and ambiguity, without confining to specific details, to foster diversity in generation. To this end, we deploy a model with *sketch* conditioning, offering strong flexibility, user-friendliness and diversity. Since all desired objects are not necessarily drawn in the sketch, *object entities* are also provided as input to complement the sketch and extract cues from the knowledge base. As depicted in Fig.3(a), the knowledge-enhanced sketch is integrated via multi-head attention. We employ a spectrum-filter (SF) to enhance meaningful object features. The conditional denoising process iteratively predicts noise for scene matrix generation (Fig.3(c)). Once the scene matrix is generated, the complete scene is decoded into a spatial field through data pop-out decoding along the object number dimension (Fig. 3(b)). Meaningless padding zeros are discarded.

Sketch serves as our primary medium for convenient user interaction, however, it lacks details to provide precise instructions for the scene generation. This aligns well with our objective of allowing diversity in generation while remaining faithful to user specifications. To enhance the instructions contained in a sketch, we integrate external knowledge that clarifies vague information. For example, when faced with a sparse sketch depicting either a "*table-aligned-sofa*" or a "*stool-aligned-sofa*," querying knowledge can assist in determining which scenario is more likely to occur in indoor scenes.

Knowledge serves as a complement or extension to the sketch in our framework. For *visible* objects in the sketch, knowledge facilitates bidirectional validation to complement sketch descriptions. When a user provides object entities that are not visible in the sketch, knowledge helps the model to accommodate plausible object shapes based on the visible side of the relationships. In summary, knowledge enhancement provides several advantages: 1) Visible sketch enhancement: When object relations are depicted, knowledge can complement any ambiguous object descriptions in the hand-drawn sketch to enhance the plausibility of the generation. 2) Invisible description complement: If the depicted relations in the sketch do not contain the desired object entities, knowledge can accommodate the *invisible* objects by relating them to the depicted ones. For example, placing an appropriate "table" alongside a "sofa".

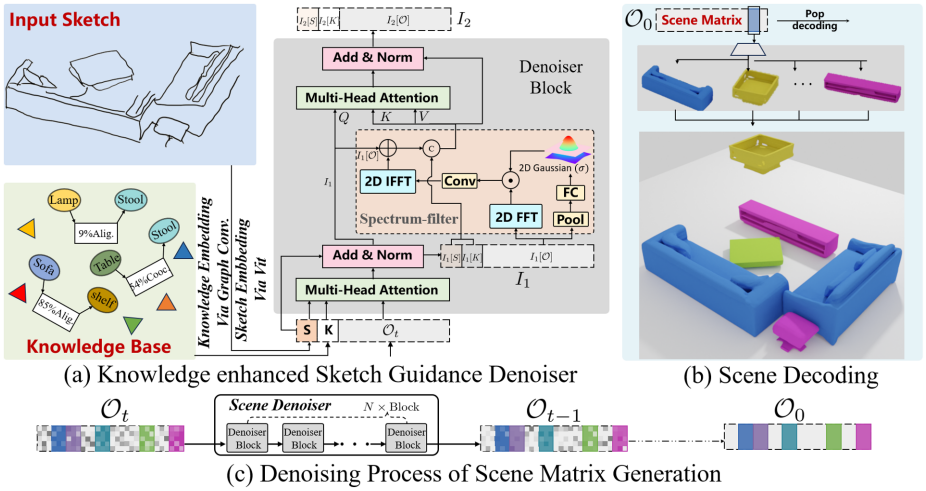

Fig. 3: Proposed SEK framework. (a) Sketch features are extracted by ViT and integrated with knowledge-enhanced reasoning features to form the denoising condition. The proposed 3D scene denoiser simultaneously generates plausible layouts and realistic 3D shapes in the matrix field. (b) The generated scene matrix is decoded to form the complete scene. (c) The denoising process: The scene denoiser starts from random noise and iteratively generates the scene matrix.

4.1 Knowledge Base

Generating a complete scene from a sketch often encounters challenges due to vague description and insufficient details. The proposed knowledge base helps in keeping the generated scenes realistic. We view knowledge as an essential complement to the scene sketch, effectively addressing its inherent sparsity and ambiguity. This section introduces our external knowledge base designed to retain extensive relationship priors for injection into the inference process. Knowledge of object relationships is first extracted from this external knowledge base and then dynamically learned to correspond with the sketch, facilitating interaction between the desired object entities.

We define knowledge base $KB = (\mathcal{V}, \mathcal{R}, p)$ as a repository containing a set of triplet relations where $\mathcal{V}$, $\mathcal{R}$, and p denote object nodes, their inner relation edges, and their edge probability respectively. The nodes $\mathcal{V}$ consist of a set of object types $f = \{f_1, f_2, \cdots, f_N\}$ and the relation edges $\mathcal{R}$ contain multiple predefined relations. Each triplet indicates the probability of the given relation existing between objects. This information is considered as external knowledge to reveal object relationships and then enhance the sketch descriptions in our framework. The predefined edge relationships $\mathcal{R}$, which include knowledge cues, are initially extracted from all indoor scenes in the given dataset. These relationships are then normalized and stored in a unique knowledge graph, forming the designed external knowledge base. Here, we introduce our scheme details to extract the priors of object relations, which comprise the external knowledge base KB.

Object Relationship Construction: External knowledge integration aids in generating reasonable semantic entities and their relationships. Hence, we extract

multiple knowledge relations $\mathcal{R}$ from the indoor scenes dateset. For generating this knowledge relation, each scene is divided into smaller functional groups using the density-based clustering algorithm, DBSCAN [3]. Initially, each object in the scene is voxelized, and then clustered into groups.

Overall, for a 3D scene S, we cluster it as $S = \{\langle \mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_K \rangle \| \mathbf{g}_i = (o_i, \ldots, o_j)\}$, where $\mathbf{g}_i$ represents the i^{th} clustered group comprising variable number of objects. Inspired by [14, 25, 49], the extracted relationships between paired objects are categorized as adjacent relations within the same group $\mathbf{g}_i, \mathbf{g}_j, i = j$:

$$\mathcal{R}_a = \{\textit{“Attachment”}, \textit{“Alignment”}, \textit{“Dependent”}\} \tag{6}$$

and distant relations cross different groups $\mathbf{g}_i, \mathbf{g}_j, i \neq j$:

$$\mathcal{R}_d = \{\textit{“Co-occurrence”}, \textit{“Parallel Collinearity”}\} \tag{7}$$

Unlike previous works that focus on object-wise relationships, we extract multiple relations to depict how entities are organized. These relations are defined as follows: (a) *Attachment*: The minimum distance between two adjacent object voxels is smaller than voxel length. (b) *Alignment*: Any plane of bounding boxes from two adjacent objects is coplanar. (c) *Dependent*: Adjacent pairwise relations in the same group that do not belong to attachment and alignment. (d) *Parallel Collinearity*: Horizontal axes of bounding boxes of the objects of different groups are parallel. (e) *Co-occurrence*: Identifying two objects that co-occur in the same scene of different groups.

Relationship Probability Counting: We count the number n_{ij} of these relationships using a clustered scene structure derived from the 3D-FRONT Dataset [12] to perform knowledge initialization. i, j denote indices of the entities, such as *chair*, *desk*. Each probability p is normalized by:

$$p_{ij} = 1/1 + e^{-10 \cdot n_{ij}^{\mathcal{R}} / max(n)^{\mathcal{R}}}. \tag{8}$$

The defined relationship edges $\mathcal{R}$ and their corresponding probability p are then integrated into the built-in knowledge base KB. The proposed SEK learns the object correlations among the given objects types, enabling it to dynamically interact with the knowledge base and assist in the diffusion module for scene generation. We build the knowledge base from 5,754 scenes in the dataset. More details are in the supplementary material.

4.2 Knowledge-enhanced Graph Reasoning

The constructed knowledge base effectively reveals the potential spatial relationships among various pieces of objects. Based on the desired object types (*chair, table, desk, etc.*), we propose a knowledge-enhanced graph reasoning module (KeGR) to incorporate external knowledge from the initialized knowledge base for comprehensive room generation reasoning. For object types $\{f_i\}_{i=1}^n$ that the scene demands, we initialize each object representation h_i of f_i via GloVe so that $h_i \in \mathbb{R}^{1 \times D_\omega}$. Next, we obtain a subset of $\{f_i\}_{i=1}^n$ to construct a fully connected subgraph $G_i^E = (h_i^E, \mathcal{E}_i^E, \mathcal{P}_i^E) \in \{KB\}$. $h_i^E \in \mathbb{R}^{n \times D_\omega}$ denoting its node feature matrix. We represent edge and probability $\mathcal{E}_i^E, \mathcal{P}_i^E$ as the initial adjacency matrix $A_i^E \in \mathbb{R}^{n \times n}$. Knowledge-enhanced graph reasoning is achieved via multistep graph convolutions:

$$H_i^{E(j)} = \delta\left(A_i^E H_i^{E(j-1)} W^{E(j)}\right), \tag{9}$$

where j denotes the j^{th} step of graph reasoning and δ is the activation function. $W^{E(j)}$ is a learnable parameter, and $H^{E(j)}$ is the node feature matrix of G^E at j_{th} step. After J iterations, we term $H_i^{E(j)}$ as the final node feature matrix of the graph reasoning of the current relationship. With all relation feature update, we get a final feature matrix group $\{H_i^{E(J)}\}|_{i=0}^{r}$, where r is the index related to relationships (6,7) of knowledge base. We perform a 1×1 convolution across the relationship dimension to get the graph feature that is used to condition the scene diffusion:

$$H^G = \delta(conv_{1\times1}(H_0, H_1, \ldots, H_r)). \tag{10}$$

4.3 Knowledge enhanced Sketch Guided Denoiser

The denoiser serves as the key module of the scene diffusion model. It predicts the noise ϵ_θ for denoising process, thereby enabling the iterative generation of the 3D scene using Eq. 5. With the external knowledge, our KeGR module produces a rich feature representation for guiding scene generation. Given the specified entities, very diverse scenes can possibly be generated. Hence, to guide the generation process w.r.t. user alignment, we include a sketch as a complementary description. Sketches are highly expressive, inherently capturing subjective and fine-grained visual cues. Furthermore, its advantage lies in the combination of easy access and vivid description. Conditioned on sketch based knowledge reasoning features, the proposed SEK denoises the 3D scene from a random point in Gaussian distribution. We employ ViT [9] as our sketch embedding backbone to obtain the sketch condition H^S maintaining the details. The conditional feature c is formed as the concatenation $[H^S, H^G] \in \mathbb{R}^{D\times2}$ of sketch and graph features.

In the forward chain of scene diffusion, the scene representation matrix $\mathcal{O}^{D\times M}$ (along with the padding) is diffused by adding Gaussian noise. As depicted in Fig. 2, padding occupies a significant portion of the scene matrix, potentially overwhelming the object information as the noise level increases. Padding is introduced in scene representation to make its dimension fixed during training. However, during inference, there is no mask available to filter out any padded values that are generated but have no specific meaning and overwhelm the desired shapes. The absence of a padding mask and the unavailability of the number of generated objects make it difficult to filter out disturbance components effectively. To address this problem, we propose component enhancement through a spectrum-filter with the intention to filter out the padding, ensuring that the prediction receives sufficient information from the valid object components. Compared to the valid object representations, we observe that the padding zeros have a low-frequency variance distribution. Note that this low-frequency distribution vanishes as noise is systematically added to $\mathcal{O}_0$ step-by-step, following the Markov chain assumption, and finally reaches the single-kernel Gaussian distribution. We apply a high-pass filter to suppress the low-frequency padding in the spectral domain. Let $\mathcal{O}_I$ denote the output of the attention blocks; the proposed spectrum-filter is computed as

$$\mathrm{EF}(\mathcal{O}_I, B) = \mathcal{O}_I + e^{-t}\Theta_{IFFT}\left(\mathrm{Conv}\left(\sigma(\mathcal{O}_I, B) \circledast \Theta_{FFT}(\mathcal{O}_I)\right)\right),$$

where t is the time step and $\circledast$ denotes high-pass filtering with adaptive Gaussian smoothed filters $\sigma(\mathcal{O}_I, B)$ (with bandwidth B), which has the same spatial size as $\mathcal{O}_I$. Θ denotes the spectrum operation using Fourier transform. Following [31], we create an initial 2D Gaussian map based on bandwidth B and apply the predefined weight parameter associated with time step t to scale the filter.

Our spectrum-filter block enhances meaningful object features in the encoded scene representation. As shown in Fig. 3(a), the scene transformer performs feature embedding first to get the embedding context initialization $\mathcal{I} = [c, t, \mathcal{O}_t] \in \mathbb{R}^{D\times(M+3)}$, where $t \in \mathbb{R}^{D\times 1}$ is the time step embedding and $\mathcal{O}_t \in \mathbb{R}^{D\times M}$ is the scene representation at time t. We begin by applying multihead attention at dimension M to capture the relevance of each element to every other element in the sequence. For example, we explore the guidance correlation between sketches and knowledge, the relevance of guidance among different conditions and every piece of object, as well as the interactions among different pieces of objects:

$$I_1 \in \mathbb{R}^{D\times(M+3)} = Atten_1(Q_1 = K_1 = V_1 = \mathcal{I}). \tag{11}$$

Next, we encode $\mathcal{I}_1$ using a transformer encoder to enrich its semantic information for each instance and then follow it by the proposed spectrum-filter block:

$$I_2 \in \mathbb{R}^{D\times(M+3)} = Atten_2(Q_2 = I_1, K_2 = V_3 = I_1[S] \copyright I_1[K] \copyright EF(I_1[\mathcal{O}])), \tag{12}$$

where, $\copyright$ denotes concatenation. Finally, we sample ϵ_{t-1} with dimensions congruent to those of scene $\mathcal{O}$ for prediction, using a set of regressive steps, where ϵ_{t-1} is the current predicted noise using the scene denoiser.

Overall, the 3D scene denoiser takes the context embedding $\mathcal{I}$ as input to perform spatial self-attention using the multi-head attention block. It is then fed to the spectrum-filter to enhance the features of valid objects and suppress invalid padding. Finally we take the output with dimension congruent to $\mathcal{O}_t$ as the predicted noise ϵ_{t-1} for supervision.

5 Experiments

Datasets: We train and test our method on three downstream tasks: 3D scene generation, 3D scene completion, and knowledge transfer validation. For the generation task, we use three types of indoor rooms from the 3D-FRONT dataset [12], including 4041 *Bedrooms*, 900 *Diningrooms*, 813 *Livingrooms*. We randomly split the data into training-test sets at 80-20% ratio. To acquire sketch, we first render images from 21 views using BlenderProc from each 3D scene uniformly when the viewpoint axis is $z_{vp} > 0$. We then apply Canny edge detection [27] to the rendered scene images to acquire their edge sketches. We manually remove the walls so that each scene contains hand-drawn looking sketches of the object. In scene completion, we randomly mask 30-80% of the objects in the scene and render it to acquire the sketches with viewpoint the same as in generation task following the above rendering process. Finally, we test the effectiveness of knowledge transfer by transferring knowledge from the 3D-FRONT dataset to the ScanNet dataset [6]. ScanNet is a real indoor scene dataset with 1,513 rooms of 21 different types. Common categories between ScanNet and 3D-FRONT dataset are selected for our knowledge transfer experiment. We retrieve objects

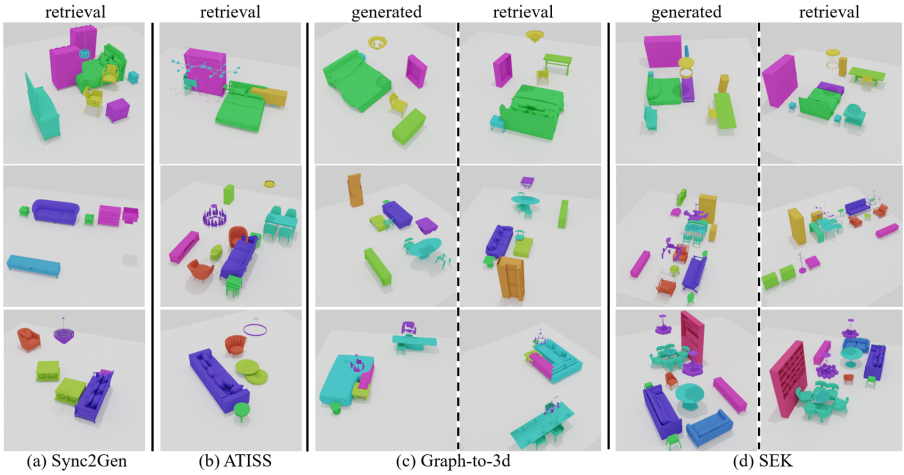

Fig. 4: Qualitative comparison. Syn2Gen and ATISS perform retrieval using 3D bounding boxes. Graph-to-3d and our method perform generation but we also show the corresponding retrieval results by searching nearest neighbor of shape code for comparison. Our method performs higher quality generation with detailed shapes and better plausibility of relationships.

of ScanNet from ShapeNet [4] to acquire consistent objects across scenes to maintain the same setting as in 3D-FRONT.

Baselines: We compare with state-of-the-art scene generation methods which can be categorised into retrieval-based and generation-based methods. In the former category, current works focus on the 3D scene layout generation followed by object placement using shape retrieval to form the complete scenes. We select the major floor plan [38, 40, 50], room size [56], and graph [29] based scene generation methods. Besides, some unconditional generation methods [46, 53] are also included for comparison. In the latter category, methods generate both shape and layouts to directly form the 3D scenes. We select the graph [7] based and unconditional [34] generation methods for comparison. For a fair comparison, we ensure the training data of baselines is the same and that each model has its required modality. Furthermore, we also compare with the most relevant sketch based method, Sketch2Scene [52]. The ATISS and Sceneformer are adopted to accept module plugin of our sketch condition for comparison.

Evaluation Metrics: Following previous works [38, 50, 53], 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. Additional information regarding evaluation metrics can be found in the supplementary materials.

5.1 Comparisons with State-of-the-art Methods

Generative Quality Evaluation: Table 1 compares the indoor scene generation quality of our method with existing state-of-the-art. Only our method performs (single view) sketch and knowledge guided 3D scene generation. Among the unconditional methods in Table 1, the diffusion based DiffuScene [46] achieves

better performance than Sync2Gen [53]. Although graph-based methods perform well on individual object generation (Fig 4(c)), these methods require a complete scene graph description that still does not specify the relative locations of objects. Hence, graph-based methods do not perform well in complete scene generation and deviate significantly from the target scene. Graph-to-Box, a variant of Graph-to-3D, focuses only on learning the object layout. Image based methods synthesize scenes under strict 2D-3D consistency. ScenePrior [34] achieves better CKL, indicating the accuracy of object classes. Layout-based methods, such as ATISS [38], start from a given layout, often a top-down wall rendering image, and perform better in terms of generation diversity and quality. However, they do not always generate reasonable scene results. Our SEK outperforms current state-of-the-art methods in quality evaluation and achieves 17.41% FID, 3.63% SCA, and 37.18% CKL better than the nearest competitor DiffuScene [46] in the Dining Room category. Figure 4 shows a qualitative comparison between Sync2Gen, ATISS, Graph-to-3D, and SEK .

Sketch based Generation Evaluation: Table 2 compares the quality of scene generation from sketches. As there are no previous generative methods specifically designed for sketch-based generation, for a fair comparison, we adopt the current state-of-the-art methods, ATISS and Sceneformer, that allow the plugin of additional conditions. We append the additional sketch to the original attention modules of ATISS and Sceneformer and refer to them as ATISS-S and Sceneformer-S, where S indicates the addition of the sketch condition. We further augment these methods by concatenating the knowledge-enhanced sketch condition, resulting in the baseline models ATISS-SK and Sceneformer-SK respectively. Additionally, we compare with the most relevant prior work in sketch-based scene synthesis, Sketch2Scene [52]. Sketch2Scene optimizes scenes to closely resemble examples in a repository while adhering to constraints from input sketches. This is done through sketch-based co-retrieval and co-placement of 3D models, ensuring similarity to reference scenes while maintaining originality. In the implementation of Sketch2Scene, since our sketch is included as a whole and lacks related pixel class information, we employ DBSCAN to manually cluster the object sketches as required in the inference stage. Note that while Sketch2Scene is the most relevant prior work in sketch-based scene gener-

Table 1: Comparative results on the 3D-FRONT dataset. For Scene Classification Accuracy (SCA), a score closer to 50% is better as it means that the generated distribution is closer to target distribution.

Method	Bedroom			Dining room			Living room		
	FID ↓	SCA %	CKL ↓	KID ↓	SCA %	CKL ↓	KID ↓	SCA %	CKL ↓
DepthGAN [56]	40.15	96.04	5.04	81.13	98.59	9.72	88.10	97.85	7.95
Sync2Gen [53]	31.07	82.97	2.24	46.05	88.02	4.96	48.45	84.57	7.52
ATISS [38]	18.60	61.71	0.78	38.66	71.34	0.64	40.83	72.66	0.69
DiffuScene [46]	18.29	53.52	0.35	32.60	55.50	0.22	36.18	57.81	0.21
Graph-to-3D [7]	61.24	74.03	1.79	54.11	76.18	1.68	41.13	79.37	2.04
Graph-to-Box [7]	55.28	69.48	1.02	50.29	73.25	1.42	48.77	78.41	1.81
3D-SLN [29]	58.17	71.27	1.38	49.67	75.39	1.44	47.29	76.29	1.77
ScenePrior [34]	24.88	83.26	0.43	46.25	89.27	0.58	44.28	88.07	0.31
FastSynth [40]	31.89	83.61	2.40	51.26	90.12	5.26	57.22	88.21	6.27
Sceneformer [50]	33.61	85.38	1.86	61.08	85.94	5.18	63.54	90.20	3.13
Ours	**15.21**	**51.24**	**0.18**	**25.46**	**51.78**	**0.16**	**31.24**	**52.91**	**0.15**

Table 2: Results for 3D scene generation based on a given sketch (and knowledge).

Method	Bedroom			Dining room			Living room		
	FID ↓	KID ↓	SCA %	FID ↓	KID ↓	SCA %	FID ↓	KID ↓	SCA %
Sceneformer-S [46]	37.21	13.04	88.37	64.38	14.21	87.16	65.78	15.03	91.14
ATISS-S [46]	21.33	3.87	64.37	42.53	6.97	74.28	44.24	7.29	77.61
Sceneformer-SK [46]	24.68	8.46	81.66	52.07	8.78	80.33	59.43	10.48	86.91
ATISS-SK [46]	18.47	1.58	58.37	37.24	2.47	63.05	39.10	3.08	63.35
Sketch2Scene [38]	22.47	7.18	55.78	41.35	6.91	72.38	58.79	7.27	84.81
Ours	**15.21**	**1.12**	**51.24**	**25.46**	**0.49**	**51.78**	**31.24**	**0.71**	**52.91**

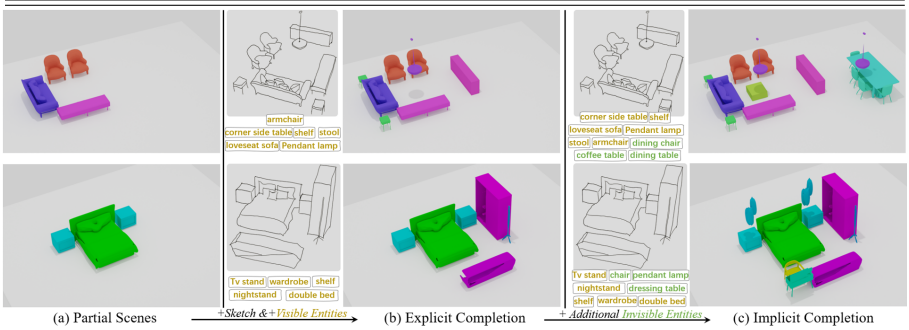

Fig. 5: Demonstration of sketch & knowledge guided scene completion. In explicit completion, the sketch and user-specified entities complement each other. Beyond explicit instructions, the additional invisible entities are inferred based on knowledge and the current visible objects to generate plausible extra objects in the scene.

ation, it necessitates an additional 3D model repository. In contrast, our method generates scenes in an end-to-end manner, without the need for such repositories.

Table 3: Results for 3D scene completion from random initial 3-5 objects.

Method	Bedroom			Dining room			Living room		
	FID ↓	KID ↓	SCA %	FID ↓	KID ↓	SCA %	FID ↓	KID ↓	SCA %
ATISS [38]	30.54	2.38	26.73	42.65	8.32	43.99	45.39	8.08	41.26
DiffuScene [46]	27.32	1.92	40.30	40.99	6.31	49.06	43.72	8.37	46.48
Ours	**21.84**	**1.58**	**45.47**	**33.03**	**5.18**	**49.45**	**35.74**	**6.51**	**48.76**

Table 4: Module ablation study

Sketch/Knowledge	SF	FID ↓	SCA %	CKL ↓
×/✓	✓	32.26	58.31	0.61
ResNet50/×	✓	34.76	58.94	1.07
ViT/×	✓	33.29	56.81	0.85
ResNet50/✓	✓	24.68	52.70	0.18
ViT/✓	×	25.83	54.19	0.37
ViT/✓	✓	**23.97**	**51.97**	**0.16**

Scene Completion: We compare against ATISS [38] and DiffuScene [46] for scene completion. For our SEK and DiffuScene, we apply the DDIM inversion process, akin to image inpainting [41], to the scene representation of the known furniture. A partial scene is obtained by the learned reverse chain following Eq. 2, and is then combined with the known scene to form the completed scene. More specifically, we retrain the model by randomly masking furniture in the sketch and test the scene completion performance. Results are given in Table 3 which show that our method performs the best on all metrics achieving average 19.12%FID, 20.06%KID, 2.61%SCA better than the nearest competitor DiffuScene [46]. Fig. 5 shows qualitative results.

Ablation Study: We perform ablation study on the condition modules to verify their contributions in Table 4. We employ ResNet50 [16] and ViT [9] with 8 at-

tention blocks and 8 attention heads as our sketch encoder. Sketch alone can not achieve good performance and performs worse than using only knowledge guidance. In the absence of knowledge, ViT shows some improvement over ResNet, but when knowledge is present, the enhancement from the sketch encoder type (ViT vs ResNet) becomes minimal. We also drop SF module for comparison (6th row). We show the More importantly, sketch and knowledge base complement each other really well and significantly improve performance when working together to jointly promote the overall quality of generation.

5.2 Knowledge Transfer to ScanNet

Knowledge transfer study is conduct to evaluate the effectiveness of the knowledge base across datasets. In our architecture, the sketch guides the spatial distribution of the objects, while the knowledge helps establish their relationships and resolves ambiguities in the sketch to generate plausible scenes. As shown in Table 5, we compare three baseline implementations of knowledge base: 1) Empty: We use an empty relationship knowledge base (parameters set to zero). 2) 3DFRONT: We directly use the external knowledge base constructed on 3D-FRONT for generation on ScanNet. 3) ScanNet: We re-train the knowledge base on ScanNet and then use it for generation on ScanNet. As expected, without relationship knowledge base, the results are much worse than when knowledge base is used. Interestingly, the knowledge extracted on 3DFRONT generates ScanNet scenes (row 2) as good as when knowledge is extracted from ScanNet itself to generate ScanNet scenes (row 3) with a very minor drop in performance on all metrics i.e. 0.34 FID, 0.02 KID, 0.95%SCA, and 0.01 CKL. This shows that our constructed knowledge base effectively transfers across datasets.

6 Conclusion

We proposed a novel sketch based knowledge-enhanced diffusion method for generating customized, diverse, and plausible 3D scenes. Our method conditions the denoising process with a hand-drawn sketch of the required scene and cues from object relationship knowledge. Given the strong generative ability of the base diffusion model, our method can take a hand-drawn sketch along with entity information to generate diverse scenes that align well with user specifications. We introduced a new condition for generation that incorporates external knowledge graphs, consisting of a set of well-defined relationship tuples. External knowledge helps resolve ambiguities for visible objects and their relationships in the hand-drawn sketches as well as introduce additional objects that are specified entities but not drawn in the sketch. Experimental results demonstrate that our model achieves state-of-the-art performance in 3D scene generation and shows promising results for the task of 3D scene completion as well.

Table 5: Knowledge transfer to ScanNet dataset.

KB Source	FID ↓	KID ↓	SCA %	CKL ↓
Empty	44.27	3.08	75.30	1.54
3DFRONT	33.81	0.83	55.18	0.18
ScanNet	**33.47**	**0.81**	**54.23**	**0.17**

Acknowledgement

This research was supported by National Key R&D Program of China under Grant 2023YFB4704800, National Natural Science Foundation of China under Grant 62293512, 62373293, 62293515, 62203160, and by ARC Discovery Project DP240101926. Ajmal Mian is the recipient of an ARC Future Fellowship Award (project number FT210100268) funded by the Australian Government.

References

1. Armeni, I., He, Z.Y., Gwak, J., Zamir, A.R., Fischer, M., Malik, J., Savarese, S.: 3d scene graph: A structure for unified semantics, 3d space, and camera. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5664–5673 (2019)
2. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data. In: international semantic web conference. pp. 722–735. Springer (2007)
3. Campello, R.J., Kröger, P., Sander, J., Zimek, A.: Density-based clustering. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **10**(2), e1343 (2020)
4. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
5. Chattopadhyay, A., Zhang, X., Wipf, D.P., Arora, H., Vidal, R.: Learning graph variational autoencoders with constraints and structured priors for conditional indoor 3d scene generation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 785–794 (January 2023)
6. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017)
7. Dhamo, H., Manhardt, F., Navab, N., Tombari, F.: Graph-to-3d: End-to-end generation and manipulation of 3d scenes using scene graphs. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16352–16361 (2021)
8. Ding, Y., Yu, J., Liu, B., Hu, Y., Cui, M., Wu, Q.: Mukea: Multimodal knowledge extraction and accumulation for knowledge-based visual question answering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5089–5098 (2022)
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. ICLR (2021)
10. Feng, M., Hou, H., Zhang, L., Guo, Y., Yu, H., Wang, Y., Mian, A.: Exploring hierarchical spatial layout cues for 3d point cloud based scene graph prediction. IEEE Transactions on Multimedia (2023)
11. Feng, M., Hou, H., Zhang, L., Wu, Z., Guo, Y., Mian, A.: 3d spatial multimodal knowledge accumulation for scene graph prediction in point cloud. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9182–9191 (2023)

12. Fu, H., Cai, B., Gao, L., Zhang, L.X., Wang, J., Li, C., Zeng, Q., Sun, C., Jia, R., Zhao, B., et al.: 3d-front: 3d furnished rooms with layouts and semantics. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10933–10942 (2021)
13. Gao, C., Chen, J., Liu, S., Wang, L., Zhang, Q., Wu, Q.: Room-and-object aware knowledge reasoning for remote embodied referring expression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3064–3073 (2021)
14. Gao, L., Sun, J.M., Mo, K., Lai, Y.K., Guibas, L.J., Yang, J.: Scenehgn: Hierarchical graph networks for 3d indoor scene generation with fine-grained geometry. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
15. Gu, J., Zhao, H., Lin, Z., Li, S., Cai, J., Ling, M.: Scene graph generation with external knowledge and image reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1969–1978 (2019)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
17. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems **33**, 6840–6851 (2020)
18. Hou, Y., Zhu, X., Ma, Y., Loy, C.C., Li, Y.: Point-to-voxel knowledge distillation for lidar semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8479–8488 (2022)
19. Hughes, N., Chang, Y., Carlone, L.: Hydra: A real-time spatial perception system for 3d scene graph construction and optimization. arXiv preprint arXiv:2201.13360 (2022)
20. Jin, A., Fu, Q., Deng, Z.: Contour-based 3d modeling through joint embedding of shapes and contours. In: Symposium on interactive 3D graphics and games. pp. 1–10 (2020)
21. Johnson, J., Gupta, A., Fei-Fei, L.: Image generation from scene graphs. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1219–1228 (2018)
22. Jyothi, A.A., Durand, T., He, J., Sigal, L., Mori, G.: Layoutvae: Stochastic scene layout generation from a label set. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9895–9904 (2019)
23. Kong, D., Wang, Q., Qi, Y.: A diffusion-refinement model for sketch-to-point modeling. In: Proceedings of the Asian Conference on Computer Vision. pp. 1522–1538 (2022)
24. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. International journal of computer vision **123**, 32–73 (2017)
25. Li, M., Patil, A.G., Xu, K., Chaudhuri, S., Khan, O., Shamir, A., Tu, C., Chen, B., Cohen-Or, D., Zhang, H.: Grains: Generative recursive autoencoders for indoor scenes. ACM Transactions on Graphics (TOG) **38**(2), 1–16 (2019)
26. Li, S., Li, H., et al.: Deep generative modeling based on vae-gan for 3d indoor scene synthesis. International Journal of Computer Games Technology **2023** (2023)
27. Li, Y., Liu, B.: Improved edge detection algorithm for canny operator. In: 2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC). vol. 10, pp. 1–5. IEEE (2022)

28. Lun, Z., Gadelha, M., Kalogerakis, E., Maji, S., Wang, R.: 3d shape reconstruction from sketches via multi-view convolutional networks. In: 2017 International Conference on 3D Vision (3DV). pp. 67–77. IEEE (2017)
29. Luo, A., Zhang, Z., Wu, J., Tenenbaum, J.B.: End-to-end optimization of scene layout. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
30. Ma, R., Patil, A.G., Fisher, M., Li, M., Pirk, S., Hua, B.S., Yeung, S.K., Tong, X., Guibas, L., Zhang, H.: Language-driven synthesis of 3d scenes from scene databases. ACM Transactions on Graphics (TOG) **37**(6), 1–16 (2018)
31. Miao, B., Bennamoun, M., Gao, Y., Mian, A.: Spectrum-guided multi-granularity referring video object segmentation. arXiv preprint arXiv:2307.13537 (2023)
32. Mikaeili, A., Perel, O., Safaee, M., Cohen-Or, D., Mahdavi-Amiri, A.: Sked: Sketch-guided text-based 3d editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14607–14619 (2023)
33. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM **38**(11), 39–41 (1995)
34. Nie, Y., Dai, A., Han, X., Nießner, M.: Learning 3d scene priors with 2d supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 792–802 (2023)
35. Nie, Y., Han, X., Guo, S., Zheng, Y., Chang, J., Zhang, J.J.: Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 55–64 (2020)
36. Parelli, M., Delitzas, A., Hars, N., Vlassis, G., Anagnostidis, S., Bachmann, G., Hofmann, T.: Clip-guided vision-language pre-training for question answering in 3d scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 5606–5611 (June 2023)
37. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 165–174 (2019)
38. 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)
39. Purkait, P., Zach, C., Reid, I.: Sg-vae: Scene grammar variational autoencoder to generate new indoor scenes. In: European Conference on Computer Vision. pp. 155–171. Springer (2020)
40. Ritchie, D., Wang, K., Lin, Y.A.: Fast and flexible indoor scene synthesis via deep convolutional generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
41. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
42. Rosinol, A., Violette, A., Abate, M., Hughes, N., Chang, Y., Shi, J., Gupta, A., Carlone, L.: Kimera: From slam to spatial perception with 3d dynamic scene graphs. The International Journal of Robotics Research **40**(12-14), 1510–1546 (2021)
43. Sanghi, A., Jayaraman, P.K., Rampini, A., Lambourne, J., Shayani, H., Atherton, E., Taghanaki, S.A.: Sketch-a-shape: Zero-shot sketch-to-3d shape generation. arXiv preprint arXiv:2307.03869 (2023)

44. Shen, Y., Zhang, C., Fu, H., Zhou, K., Zheng, Y.: Deepsketchhair: Deep sketch-based 3d hair modeling. IEEE Transactions on Visualization and Computer Graphics **27**(7), 3250–3263 (2021). https://doi.org/10.1109/TVCG.2020.2968433
45. Speer, R., Chin, J., Havasi, C.: Conceptnet 5.5: An open multilingual graph of general knowledge. In: Proceedings of the AAAI conference on artificial intelligence. vol. 31 (2017)
46. Tang, J., Nie, Y., Markhasin, L., Dai, A., Thies, J., Nießner, M.: Diffuscene: Scene graph denoising diffusion probabilistic model for generative indoor scene synthesis. arXiv preprint arXiv:2303.14207 (2023)
47. Teney, D., Liu, L., van Den Hengel, A.: Graph-structured representations for visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2017)
48. Tulsiani, S., Gupta, S., Fouhey, D.F., Efros, A.A., Malik, J.: Factoring shape, pose, and layout from the 2d image of a 3d scene. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 302–310 (2018)
49. Wald, J., Dhamo, H., Navab, N., Tombari, F.: Learning 3d semantic scene graphs from 3d indoor reconstructions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3961–3970 (2020)
50. 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)
51. Wu, Z., Wang, Y., Feng, M., Xie, H., Mian, A.: Sketch and text guided diffusion model for colored point cloud generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8929–8939 (2023)
52. Xu, K., Chen, K., Fu, H., Sun, W.L., Hu, S.M.: Sketch2scene: Sketch-based co-retrieval and co-placement of 3d models. ACM Transactions on Graphics (TOG) **32**(4), 1–15 (2013)
53. 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)
54. Yang, L., Huang, Z., Song, Y., Hong, S., Li, G., Zhang, W., Cui, B., Ghanem, B., Yang, M.H.: Diffusion-based scene graph to image generation with masked contrastive pre-training. arXiv preprint arXiv:2211.11138 (2022)
55. Yang, M.J., Guo, Y.X., Zhou, B., Tong, X.: Indoor scene generation from a collection of semantic-segmented depth images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15203–15212 (2021)
56. Yang, M.J., Guo, Y.X., Zhou, B., Tong, X.: Indoor scene generation from a collection of semantic-segmented depth images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15203–15212 (2021)
57. Yin, T., Zhou, X., Krahenbuhl, P.: Center-based 3d object detection and tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11784–11793 (2021)
58. Zareian, A., Karaman, S., Chang, S.F.: Bridging knowledge graphs to generate scene graphs. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16. pp. 606–623. Springer (2020)
59. Zhang, S.H., Guo, Y.C., Gu, Q.W.: Sketch2model: View-aware 3d modeling from single free-hand sketches. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6012–6021 (2021)