Draw Step by Step: Reconstructing CAD Construction Sequences from Point Clouds via Multimodal Diffusion.

Weijian Ma Shuaiqi Chen Yunzhong Lou Xueyang Li Xiangdong Zhou*
School of Computer Science and Technology, Fudan University
{mawj22, chensq22, xueyangli21}@m.fudan.edu.cn {yzlou20, xdzhou}@fudan.edu.cn

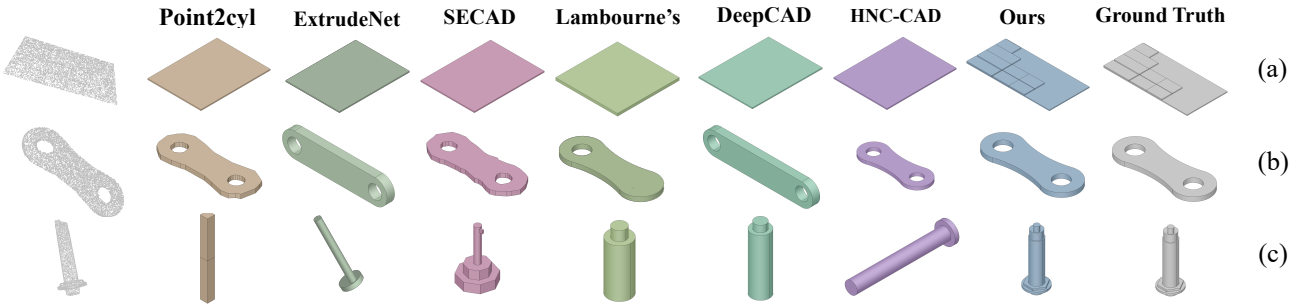


Figure 1. The reconstruction results. We render the reconstructed CAD construction sequences of our proposed CAD-Diffuser and comparative methods. Owing to the combination of our proposed multimodal token-based diffusion strategy and the volume-based noise schedule following the top-down design paradigm, our method has better grasped the information conveyed in the construction sequences and has a better understanding of the CAD geometry, leading to more detailed reconstruction results.

Abstract

Reconstructing CAD construction sequences from raw 3D geometry serves as an interface between real-world objects and digital designs. In this paper, we propose CAD-Diffuser, a multimodal diffusion scheme aiming at integrating top-down design paradigm into generative reconstruction. In particular, we unify CAD point clouds and CAD construction sequences at the token level, guiding our proposed multimodal diffusion strategy to understand and link between the geometry and the design intent concentrated in construction sequences. Leveraging the strong decoding abilities of language models, the forward process is modeled as a random walk between the original token and the [MASK] token, while the reverse process naturally fits the masked token modeling scheme. A volume-based noise schedule is designed to encourage outline-first generation, decomposing the top-down design methodology into a machine-understandable procedure. For tokenizing CAD data of multiple modalities, we introduce a tokenizer with a self-supervised face segmentation task to compress local and global geometric information for CAD point clouds, and the CAD construction sequence is transformed into a primitive token string. Experimental results show that our CAD-Diffuser can perceive geometric details and the results are more likely to be reused by human designers.

*Corresponding author.

1. Introduction

Starting from a shape flashed across their eyes and mind, human designers first quickly depict what they perceive in a draft outline. Then, they iteratively judge, discuss, and revise their designs until they become detailed parametric Computer-Aided Design (CAD) models. Finally, the completed parametric models are further refined for downstream manufacturing processes. Such a top-down design workflow constitutes most manufactured objects in the real world, from coffee mugs to airplanes. However, there does exist situations where parametric design models are not available, which makes concise and reusable reconstruction from 3D data like point clouds vital. Hence we set off to ask: *how can we leverage the inherent logic passed down by human designers to better reconstruct parametric modelling sequences from point clouds?*

Top-down design is one of the most profound design paradigms summarized by human designers. In terms of parametric CAD model design, it means that shape construction command sequences should be concise, modifiable, and reusable. It reflects the designer's intents such as the outlines are drwan first, while the introduction of meaningless entities should be minimized. There have been many previous works on conditional or unconditional parametric model generation. However, few works have attempted to take design methodology into account, especially in the field of reconstructing a parametric sequence from geome-

try. Some previous work only reconstructs models with a fixed small number of sketch-extrusion pairs [22, 29, 37]. For primitives in sketches, such methods either regard them as B-splines [29], or rely on CAD tools [37] and heuristic methods [22] to predict sketch primitives from 2D Signed Distance Field (SDF). They achieved superior geometric reconstruction accuracy but overlooked the design paradigm of human designers. Recent works [45, 49] regard CAD construction sequences as language, bringing new inspirations to the problem.

We argue that reconstructing CAD models suitable for top-down human design workflows should acquire knowledge of how humans conceive and express their designs, which is concentrated in CAD construction sequence in the form of structured language. In this sense, different from previous methods, we combine the strong generation ability of diffusion models and representation ability of language models, and propose a novel multimodal diffuser. Our idea is based on the following observations. First, point clouds and CAD construction sequences are both sequential data. The conversion between them is a multimodal translation between point cloud tokens and CAD construction sequences. Second, the top-down design paradigm of constructing CAD models from outlines to details is a stepwise modeling activity. It can be imitated by diffusion models with specially designed noise schedules.

Based on the intrinsic of point cloud data and CAD construction sequences, we adopt a new paradigm which consists of a multimodal text diffuser and a point cloud tokenizer. The multimodal diffuser is a token-based diffusion model based on masked token modeling. With the help of a volume-based noise schedule, tokens of CAD sequences are denoised from outlines to details. The point cloud tokenizer is an improved discrete VAE trained with a self-supervised face segmentation task based on metric learning. Experimental results show that the CAD models generated by our method not only surpass previous methods on the likeliness of CAD sequences, but also more diverse and likely made by human designers, due to the strong generation ability of the combination of diffusion models and language models.

To sum up, our main contributions are as follows.

- We are the first to model the reconstruction of the parametric CAD models from point clouds as a multimodal diffusion between tokens of different modalities.
- We present a multimodal diffusion method based on masked token modeling along with a novel noise schedule following top-down design paradigm of human designers.
- We propose a point cloud tokenizer trained on face segmentation pretext task via a metric learning technique.
- Experimental results on commonly used CAD parametric model datasets show that our model can generate CAD sequences with more accuracy, more generation diversity as well as higher reusability.

2. Related Work

CAD Model Generation. There are a number of attempts to generate parametric CAD models with or without conditions. For unconditional generation, a line of work focused on directly building the geometry of CAD models through generating parametric curves [41] or surfaces [32] based on fixed [33] or arbitrary sketches [43] and solid models [12, 17, 40]. The recently emerged large-scale parametric CAD datasets [20, 45] have enabled language models to simulate the design patterns of human architects. When generating with conditions, the provided conditions could be point cloud like DeepCAD [45], partial CAD input like HNC-CAD [49], target B-reps [43, 47], voxel grids like SECAD [21, 22] or point clouds with [37] like Point2cyl, or point clouds without [22, 29] sequence guidance. However, all these generation methods are one-pass or via a predefined hierarchy, leading to inferior reconstruction performance compared with diffusion models, which divide the generation procedure into a machine understandable one based on noise schedules and timesteps.

Converting Point Clouds into CAD Construction Sequences. Primitive fitting and object decomposition have long been researched in computer vision history. Heuristic-based attempts include decomposing with RANSAC [9, 31], region growing methods, or Hough transforms [4, 8]. Recently, with the advent of deep learning in 3D domains and the widespread adoption of implicit representations, a line of works try to fit sketch primitives from reconstructed implicit representations. Point2cyl [37] predicts the extrusion parameters and adopts CAD software to predict primitives based on rendered images from 2D SDF. ExtrudeNet [29] regards each sketch as a combination of B-splines and predicts their control points, while a differentiable extruder and a differentiable combiner are utilized to convert the 2D sketch SDF based on reconstructed Bezier curves into 3D extrude volumes. SECAD-Net [22] predicts sketch primitives from 2D SDF by Teh-Chin chain approximation [36] and Dierkx Fitting [7] to convert sketches into B-splines. Such works generate CAD construction sequences, but the sequences are obtained by using CAD tools, based on heuristics, or solely composed of B-splines.

Token Generation via Language Models. Generating tokens via diffusion is much harder than generation in continuous domains such as images [34] and embeddings [28, 30] due to its discrete nature [24, 54]. Most research works focus on text generation [10, 14, 53]. Existing studies on text diffusion can be divided mainly into two categories, *continuous diffusion* and *discrete diffusion*. Like image diffusion models [6, 15, 34, 35], continuous diffusion performs noising and denoising strategies in a continuous latent space by applying Gaussian noise to latent embeddings [10, 51]. The transition between text and embeddings is completed via an autoencoder. Another line of work modifies the

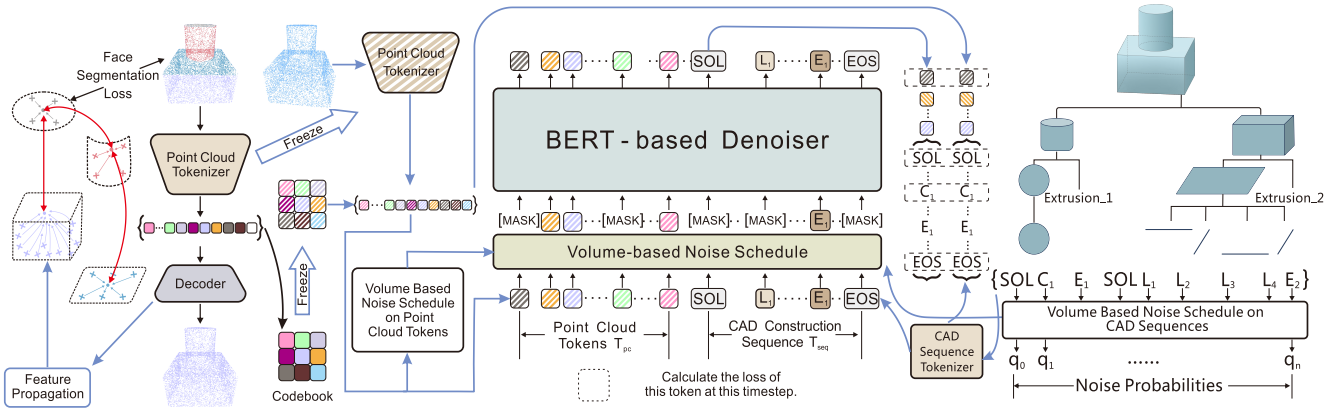


Figure 2. The overall schedule of our proposed CAD diffuser. Point clouds are first compressed into tokens. At each timestep, the masked point cloud tokens and CAD construction sequence tokens are jointly sent to the BERT-based denoiser where all masked tokens are predicted. During training, the loss is calculated via measuring the correctness of tokens that are masked in the specified timesteps. At inference time, the partially unmasked sequence will be sent to the denoiser to continue the reverse process. The hierarchy of the CAD sequence and the tokenization technique of point clouds will be introduced in Section 4.4 and Section 4.5 accordingly.

diffusion models for text data by introducing discrete forward processes like categorical transition kernels [16], absorbing kernels [2, 5, 39] and uniform transition kernels [10, 23]. DiffusionBERT [14] and Diffusion-NAT [53] have integrated the diffusion process into the Masked Language Modeling scheme by regarding noise addition as masking a word. Such works fully utilize the pretrained language models as well as their training paradigms. VQ-Diffsion [11] compresses images via VQVAEs [38] and models the diffusion process as a random walk through codebook entries. LayoutDiffusion [52] models the diffusion process as a mixed strategy in which some tokens have absorbing kernels while others move freely. However, no previous work extends token-based diffusion to multimodal domains.

3. The Overall Schedule

The problem of CAD sequence reconstruction from point cloud can be formulated as *translation* between *point cloud* tokens T_{pc} and *primitives tokens* of CAD construction sequences T_{seq}, as shown in Figure 2. We aim to integrate the reverse engineering CAD construction sequence into the top-down design paradigm of human designers, which requires a deep understanding and linking between the geometry and the design knowledge highly concentrated in the sketch-extrusion pairs of the construction sequences.

To achieve this, we propose CAD-Diffuser, a novel multimodal diffusion strategy that unifies the source modality and the target modality at the token level, since tokens of CAD construction sequences possess exact meanings of design. The forward process and the backward process are modeled as masking and unmasking tokens in the masked token modeling scheme commonly used in prevailing language models like BERT [18], as shown in Figure 3.

The overall pipeline of the CAD-Diffuser is shown in Figure 2. Before training our multimodal scheme, a point cloud tokenizer is trained in a novel self-supervised manner based on a face segmentation pretext task. For our multimodal diffuser, given a point cloud of the CAD model, it is first tokenized into T_{pc} through the pretrained point cloud tokenizer. T_{pc} is then sent to the multimodal diffusion scheme as a condition to maximize the posterior probability $p(T_{seq}|T_{pc})$. During training, the volume-based noise schedule is used to calculate the noise probability in the forward process. This instructs the model with the top-down design paradigm where the details are masked out first and finally the outlines. In the reverse process, the denoiser predicts all tokens at one time, while only the noises, namely the masked tokens, added at the exact timestep t will be calculated. At inference time, the unmasked sequence will be sent back to the denoiser to make further predictions until the timestep reaches 0. The details of each part will be illustrated in the following sections.

4. Details of Multimodal Token-based Diffuser

In contrast to recently proposed multimodal diffusion models [28, 30] where multimodal diffusion happens at continuous latent space, we instead introduce the multimodal diffusion in input space at the discrete token level, as shown in Figure 3. The design choice is based on the following considerations. First, point clouds and CAD construction sequences are both discrete data suitable for tokenization. Moreover, modality interaction at token level can exert fine-grained token-wise control where each token in one modality can interact with any token in its own modality and the other. Each CAD sequence token can refer to the information of point cloud tokens at arbitrary level and vice versa.

4.1. Diffusion Models

Diffusion models are generative models that work by perturbing training data through successive addition of designed noise, and then learning to recover the data by reversing this noising process to acquire generation ability. A

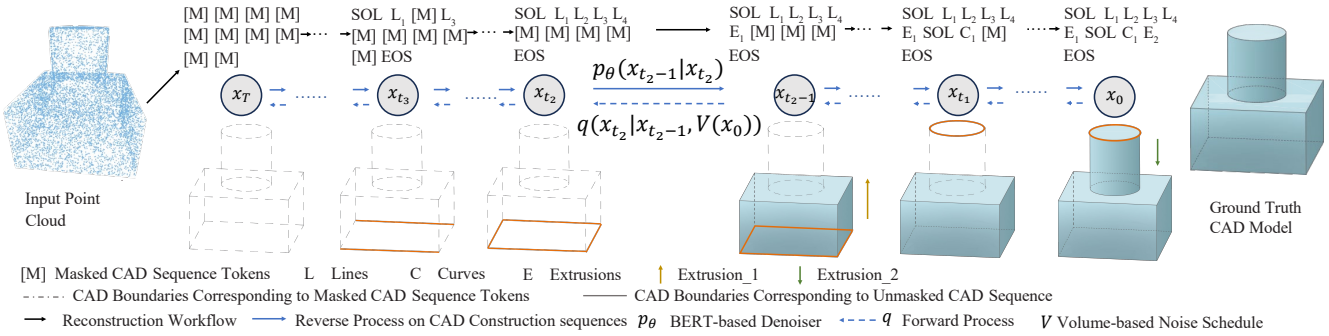


Figure 3. The reconstruction process. Our proposed CAD-Diffuser takes a CAD point cloud as input and converts it into CAD construction sequence, which is generated step by step via our proposed multimodal diffusion strategy. The diffusion strategy is a random walk between the original token and [MASK] token, which naturally adapts to denoising language models like BERT. The diffusion process is guided by our proposed volume-based noise schedule, which imitates the top-down design paradigm where outlines are drawn first and details are carved later, as is illustrated by the renderings of the CAD sequences shown in the second line.

formal definition of diffusion models [6, 15, 34] is as follows. For a sample $x_0 \sim q(x_0)$, latent variables $x_1, \cdots, x_T$ are generated through a forward process by adding a small Gaussian noise to the sample

$$q(x_t|x_{t-1}) = N(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I), \tag{1}$$

where $\{\beta_t \in (0,1)\}_{t=1}^T$ is a noise schedule controlling the noise added until x_T becomes a Gaussian distribution eventually. The reverse process $q(x_{t-1}|x_t)$ can also be regarded as Gaussian when β_t is small enough, which can be represented as

$$p_\theta(x_{t-1}|x_t, t) = N(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \tag{2}$$

where $\mu_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ can be networks. The reverse process can also condition on x_0, where $q(x_{t-1}|x_t, x_0)$ has a closed form. The training target is to minimize the variational lower bound to optimize $\log p_\theta(x_0)$:

$$\begin{aligned} &L_{vlb} = E_q[D_{KL}(q(x_T|x_0)\|p_\theta(x_T))] \\ &+E_q[\textstyle\sum_{t=2}^T D_{KL}(q(x_{t-1}|x_t, x_0)\|p_\theta(x_{t-1}|x_t, t))] \\ &-\log p_\theta(x_0|x_1), \end{aligned} \tag{3}$$

where E_q is the expectation over joint distribution $q(x_{0:T})$.

4.2. Multimodal Token Diffusion as a Random Walk

The discrete nature of tokens makes it difficult to apply a continuous noise schedule at token domains, since *the intermediate state* between noise and the original tokens hardly exists and most of the transitions between language tokens are not interpretable. We propose a specific instance of discrete diffusion models where the diffusion of tokens in different modalities is modeled as a random walk between the original token and [MASK] token with an *absorbing state* and a *reflective state*. A transition matrix is adopted to integrate our diffusion strategies with denoising language models like BERT. More characteristics of discrete diffusion can be found in [10, 14, 24].

We combine language model training with our multimodal diffusion strategy by regarding transition between a token and [MASK] in the training scheme of BERT [18] as the noising and denoising procedure, extending [14] to multimodal domains. For multimodal diffusion, tokens in the source modality is *known* while tokens in the target modality remain *unknown*. For tokens in the source modality, [MASK] represents a *reflective state*, where the masking of tokenized source is only for language modeling purposes, and the source tokens remain unchanged during the diffusion process. For tokens in the target modality, [MASK] acts as an *absorbing state* where the action space of each token remains unchanged or transitions to [MASK]. The formal definition of each token x^i at timestep t is as follows,

$$[Q_t]_{i,j} = \begin{cases} 1 & \text{if } i = j = [M] \quad x_0^i \in \{\text{target}\} \\ 1 & \text{if } j = x_{t-1}^i, i = [M] \quad x_0^i \in \{\text{source}\} \\ \beta_t & \text{if } j = [M], i \neq [M] \\ 1-\beta_t & \text{if } i = j \neq [M] \\ 0 & \text{others,} \end{cases} \tag{4}$$

where x_t^i denotes the i-th token at step t and $[M]$ represents the [MASK] token. The random walk converges to a stationary distribution $q(x_T)$ where all source tokens remain unchanged while all target tokens lie at $[MASK]$ with probability 1.

The closed form of marginal distribution $q(x_t^i|x_0^i)$ at step t is derived as follows,

$$q(x_t^i|x_0^i) = \begin{cases} \overline{\alpha}_t & \text{if } x_t^i = x_0^i \quad p_0 \in \{\text{target}\} \\ 1-\overline{\alpha}_t & \text{if } x_t^i = [M] \quad p_0 \in \{\text{target}\} \\ 1-\overline{\gamma}_t & \text{if } x_t^i = x_0^i \quad p_0 \in \{\text{source}\} \\ \overline{\gamma}_t & \text{if } x_t^i = [M] \quad p_0 \in \{\text{source}\} \end{cases}, \tag{5}$$

where $\overline{\alpha}_t = \prod_{i=1}^t (1-\beta_i)$, $\overline{\gamma}_t = \sum_{i=0}^t (\prod_{j=0}^i ((-1)^j \beta_{t-j}))$. In practice, $\overline{\gamma}_t$ is set to 0 at step T. In the problem setting of CAD sequence reconstruction from point clouds, the *source* modality refers to the point cloud modality while the *target* modality is the CAD construction sequence.

Similar to previous works [14, 30, 34, 49], the training target of the reverse diffusion process, namely the optimiza-

tion target of $p_\theta(x_{t-1}|x_t, t)$, is shown as follows,

$$p_\theta(x_{0:T}) = p(x_T) \prod_{\ell=1}^{T} p_\theta(x_{t-1}|x_t, t). \tag{6}$$

From the definition above, we can define the forward process of multimodal diffusion is equivalent to adding [MASK] to token sequence according to Equation 5. And the reverse process can be safely regarded as unmasking tokens thanks to the strong representation and decoding abilities of BERT [18, 25, 42] and masked data modelling [13].

4.3. Volume-based Noise Schedule on CAD Tokens

In both continuous [15] and discrete domains [14], noise schedule is shown to be critical to the performance of the diffusion model. In previous work on language diffusion, noise is added by using the uniform transition matrix [34], absorbing state transition [2], or noise schedule marking corpus word occurrence [14]. Such attempts are shown to be successful in natural language generation. However, CAD construction sequence is made up of keyword tokens, which is illustrated in the following section. It inherently conveys the design knowledge such as the top-down design paradigm and can fit into the noise schedule to guide the generation procedure. Such a noise schedule design can be extended to all sequences containing keyword tokens.

From the traits of CAD model design, we propose a noise schedule that (1) mimics the top-down design paradigm where the outlines are generated first from the masked tokens. (2) Measures the added noise at each timestep by using the proportion of volume corrupted at the timestep. (3) Tokens in a CAD sequence are categorized in descending order of their contributed volumes. In other words, we mask the tokens carving geometric details at the beginning of forward process so that the learned reverse process first draws the outline, and finally details of the model will emerge.

Formally, distributing corrupted information across the forward step can be described as follows,

$$1 - \frac{t}{T} = \frac{\sum_{i=1}^{n} V(x_t^i)}{\sum_{i=1}^{n} V(x_0^i)} = \frac{\sum_{i=1}^{n} \overline{\alpha}_t^i V(x_0^i)}{\sum_{i=1}^{n} V(x_0^i)}, \tag{7}$$

where V denotes the entropy of CAD sequence, which measures the proportion of volume a CAD sequence token occupies. x_i is the i-th token in the CAD construction sequence and n denotes the sequence length. From Equation 5, $\overline{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$ denotes the probability that $x_i^t = x_i^0$, i.e., the i th token remains unmasked at step t. Our goal is to design the outline-first generation, so $\overline{\alpha}_t^i > \overline{\alpha}_t^j$ when $V(x_t^i) < V(x_t^j)$, and the outlines will emerge before the details. In practice, for CAD sequence tokens, the proportion of a token x_i^0 in a sketch S contributing to the volume is first calculated by the length it occupies in a sketch, then multiplied by the volume the sketch-extrusion pair occupies in the whole volume. The proportion of an extrusion token E is assigned as the volume of its sketch-extrusion pair. For CAD point cloud tokens, the noise is distributed evenly.

From the properties above, $\overline{\alpha}_t$ is defined as follows,

$$\overline{\alpha}_t^i = 1 - \frac{t}{T} - S(t) \cdot \tilde{V}(x_0^i), \tag{8}$$

$$S(t) = \lambda \sin \frac{t\pi}{T}, \tag{9}$$

$$\tilde{V}(x_0^i) = 1 - \frac{\sum_{j=1}^{n} V(x_0^j)}{n V(x_0^i)}, \tag{10}$$

$$V(x_0^i) = \begin{cases} \dfrac{L(x_0^i)}{\sum_{x_0^i \in S_j} L(x_0^i)} \dfrac{V(S_j, E_j)}{\sum_{j=0}^{m} V(S_j, E_j)} & x_0^i \in S \\ \dfrac{V(S_j, E_j)}{\sum_{j=0}^{m} V(S_j, E_j)} & x_0^i \in E \\ \dfrac{1}{|T_{pc}|} & x_0^i \in T_{pc}, \end{cases} \tag{11}$$

where $S(t)$ is to control the effect of the volume-based schedule at step t. The sinusoidal implementation is to let $S(0) = S(T) = 0$ so that x_t retains all zeros at the beginning or the end of the diffusion process. λ is a hyperparameter controlling the effect of the volume-based noise schedule that the noise schedule degrades to [34] where $\beta_t = (T-t+1)^{-1}$. L represents the length of each sketch token. $V(S, E)$ represents the volume of each sketch-extrude pair. $|T_{pc}|$ represents the number of point cloud tokens.

Details of tokens in each modality will be illustrated in Section 4.4 and 4.5. For other details of the multimodal diffuser, please refer to [10, 14, 24].

4.4. The Hierarchical CAD Sequence Tokens

A CAD model can be expressed in a structured design language with hierarchical design flows. An example is illustrated in the right part of Figure 2. From the top-down perspective, a CAD model M is a Boolean combination of extruded cylinders. An extrusion cylinder is defined as a combination of a sketch S and an extrusion parameter E marking the location s of a sketch plane, the direction d of the extrusion axis, the scale l and the Boolean relationship b of the extrusion. For a sketch plane S, it is composed of a set of loops L in the same plane. A loop is a combination of different geometric primitives C that forms a closed curve. Each geometric primitive contains its type t and its corresponding parameters p.

The hierarchical representation can be summarized as a depth-first transversal of a primitive token tree. For example, the CAD model shown in the right part of Figure 2 can be represented by the following sequence $[SOL\ C_1\ E_1\ SOL\ C_2\ C_3\ C_4\ C_5\ E_2]$ where each extrusion cylinder is naturally represented by extrusion E and SOL marks as the start of a loop. Our representation is a modification of DeepCAD [45], where unnecessary parts are removed.

4.5. Point Cloud Tokenizer

CAD shapes constructed from sketch and extrusions have flat surfaces and tangent extrusion sides [45]. For learning CAD point cloud vocabularies D, the feature extractor needs to understand the local patch geometry, as well as the long-range relationship for points in the same sketch face S or extrusion side E. Specifically, following [26, 27, 50], we divide point cloud into local patches $\{p\}_{i=1}^{g}$ and project 3D patches into local embeddings e via a PointNet layer. The embeddings are then converted into discrete point tokens z stored in a codebook D with learned vocabularies. Therefore, the learned vocabularies are obtained via a discrete VAE by pretraining on all the CAD point clouds.

To preserve the sketch face and extrusion side relationship, we design a self-supervised face segmentation pretext task, extending the method of [46] to 3D domains. As shown in the left part of Figure 2, features of each reconstructed point $p' \in P'$ is projected back to the original point cloud P by inverse distance weighting. Then the projected feature on the original point cloud is clustered via a vMF Mean Shift method [3] so that points features clustered in a face will be drawn near in the embedding space while features in different faces will be torn apart.

Formally, the overall pretraining tokenization objective can be represented as $E_{z\sim Q_\phi(z|p)}[\log P_\varphi(p|z)]$ where Q stands for encoder and tokenizer, and P for decoder. The tokenization objective can be viewed as a combination between maximizing the Evidence Lower Bound [19] of the reconstructed and the original point cloud and minimizing the intra-cluster features while maximizing the inter-cluster features. The overall loss is calculated as follows,

$$L_{\text{CD}} = \frac{1}{|P'|}\sum_{p'\in P'}\min_{p\in P}\|p'-p\| + \frac{1}{|P|}\sum_{p\in P}\min_{p\in P'}\|p-p'\|, \tag{12}$$

$$\begin{aligned} L_{\text{seg}} &= L_{\text{intra}} + L_{\text{inter}} \\ &= \frac{1}{K}\sum_{k=1}^{K}\sum_{i=1}^{N}\frac{1\{d(\mu^k, x_i^k)-\alpha\geq 0\}d^2(\mu^k, x_i^k)}{\sum_{i=1}^{N}1\{d(\mu^k, x_i^k)-\alpha\geq 0\}} \\ &+ \frac{2}{K(K-1)}\sum_{k<k'}[\delta - d(\mu^k, \mu^{k'})]_+^2, \end{aligned} \tag{13}$$

$$L_{\text{tokenizer}} = L_{\text{CD}} + L_{\text{seg}} + D_{KL}[Q_\phi(z|p_i), P_\varphi(z|\tilde{p}_i)]), \tag{14}$$

where x_i^k is the aggregated feature of p_i from P' via inverse distance weighting under class k. μ_k is the center of all points with label k. K is the total number of classes. d represents the Euclidean distance. α and δ are hyperparameters controlling the distance of inter- and intra-class distances respectively.

In all, the design of our multimodal token-based diffuser serves as a natural fit to the CAD construction sequence reconstruction. The proposed diffusion strategy divides the multimodal translation process into a fine-grained controllable schedule. The noise schedule mimics top-down design paradigm and conveys the hierarchical design knowledge in a machine understandable manner. The tokenization strategy in each modality reasonably divides construction sequence and point cloud data into hierarchical tokens.

5. Experiments

In this section, we evaluate the performance of CAD-Diffuser on DeepCAD [45] and Fusion 360 Gallery [44]. The effectiveness of our approach, namely the reconstruction accuracy and the generation diversity over partial inputs, will be verified by extensive comparison with state-of-the-art methods.

5.1. Experimental Setups

Datasets. CAD-Diffuser is trained on DeepCAD training set and the experimental results on the test set of both DeepCAD and Fusion 360 Gallery are reported. Both DeepCAD and Fusion 360 Gallery dataset contain CAD construction sequences that can be rendered into geometric shapes via CAD geometry kernels [1]. The settings of the CAD construction sequences and point clouds are the same as DeepCAD except that the CAD construction sequences are represented in a hierarchical manner, see Section 4.4.

Training Details. The training is composed of two stages. In the first stage, the point cloud tokenizer is trained for 200 epochs of batch size 64 and learning rate $1e-4$ with linear warmup and cosine learning rate schedule. During this stage, underutilized codebook tokens will be randomly reinitialized like [49]. All point cloud will be tokenized into codebook entries of length 64 when the first stage of training completes. In the second stage, the multimodal diffuser is trained on a DistilBERT backbone where the tokenized codebook entries are the source and the CAD construction sequences are the target. The multimodal diffuser is trained with batch size 512 for 350 epochs under learning rate $5e-5$.

Evaluation Metrics. For quantitative evaluations, we adopt metrics that are commonly used in previous methods [45], including Command Accuracy (Acc_{ct}), Parameter Accuracy (Acc_{cp}), Chamfer Distance (CD), Intersection over Union (IoU) and Invalid Rate (IR). Additionally, we also report the difference between the number of ground truth primitives and generated primitives $\#\Delta P$. Both (Acc_{ct}), (Acc_{cp}) and $\#\Delta P$ serve as a measure of how the reconstructed CAD sequences are like the original human-designed sequences. Details of these metrics are listed in the supplementary materials.

5.2. Reconstruction Accuracy

We thoroughly compare our method with several types of CAD reconstruction methods that generates editable

Methods	DeepCAD						Fusion 360					
	Acc_{ct}	Acc_{cp}	Med CD	IoU	IR	$\#\Delta P$	Acc_{ct}	Acc_{cp}	Med CD	IoU	IR	$\#\Delta P$
Methods Based on Reconstruction												
Point2cyl†[37]	35.32%	32.74%	0.427	0.738	3.87%	28.73	37.18%	36.85%	0.418	0.675	3.22%	27.36
ExtrudeNet†[29]	28.17%	24.73%	0.337	0.403	25.34%	35.86	27.43%	23.75%	0.495	0.373	24.97%	29.17
SECAD-Net†[22]	36.71%	27.48%	0.365	0.729	7.72%	36.13	35.02%	28.24%	0.432	0.690	7.46%	29.99
Method Based on Searching												
Lambourne's‡[21]	72.96%	66.78%	0.428	0.721	16.16%	2.58	66.38%	62.76%	0.475	0.653	17.98%	9.32
Methods Based on Language Models												
DeepCAD[45]	80.39%	69.60%	0.919	0.467	15.44%	4.16	67.09%	57.65%	8.923	0.399	25.17%	6.51
HNC-CAD* [49]	82.69%	74.58%	0.864	0.653	5.62%	3.56	75.46%	64.52%	3.682	0.635	7.27%	5.24
Our Method	88.55%	82.92%	0.302	0.743	1.48%	2.20	85.56%	80.48%	0.385	0.632	1.65%	1.23

Table 1. Results on reconstruction accuracy. † indicates results converted into construction sequences via heuristic methods. ‡ indicates results obtained via re-implementing methods according to the original paper. $*$ indicates baseline methods adapted from the original paper. All metrics are tested on **the translated CAD sequences and their rendered geometry**. Median CD is multiplied by 10^2. For details of comparative methods, please refer to supplementary materials.

primitive-based CAD models. The methods can be divided into three categories, namely reconstruction-based methods [22, 29, 37], search-based methods [21], and two methods based on language models for CAD reconstruction [45, 49]. For each method, we adopt their original implementations for training and finetuning, and *converts the results into CAD construction sequences*. Such methods offer sufficient comparisons to other techniques and establish themselves as state-of-the-art. For detailed implementation of these methods, please refer to the supplementary materials.

Table 1 shows the quantitative results. It can be seen that for sequence reconstruction accuracy, our method outperforms all methods in view of command and parameter accuracy. The smallest value of $\#\Delta P$ also implies that CAD-Diffuser has fit the human designs well, which has neither scattered the primitives nor oversimplified them. For geometric indicators like CD and IoU, the results show that our method is on par with the 3D reconstruction based methods, which shows that our method reaches the sequence reconstruction accuracy without sacrificing the geometric fidelity. It is worth noting that the Invalid Rate and Median CD are not so well on Point2cyl, ExtrudeNet and SECAD-Net. This is because the sketch approximation heuristics scarifies accuracy and triggers exceptions in CAD kernels [1].

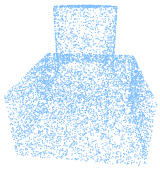
Figure 1 shows some typical qualitative reconstruction results. As shown in the figure, the geometric fidelity of our method is on par with other methods, while the details of the CAD models are fully recovered by our method, especially in the first line where the rectangles on the top are hardly perceptible in the point clouds. This strongly shows that our token-based multimodal diffusion schedule can bring about fine-grained preception of geometry and generation controls.

Method	N-gram (↑)	Embedding Similarity(↓)	Edit Distance(↑)	COV (↑)	MMD (↓)
DeepCAD	0.18	0.193	0.44	80.62	1.10
SkexGen	0.21	0.175	0.47	84.74	1.02
HNC-CAD	0.35	0.138	0.56	87.73	0.96
Ours	0.45	0.082	0.73	88.35	0.94

Table 2. Results on generation diversity. We report distinct token-level **N-grams**, pairwise similarities of DeepCAD **embeddings**, pairwise normalized **Edit Distance** of generated sequences, test set *Coverage* (**COV**) of generated CAD sequences and *Minimum Matching Distance* (**MMD**) between generated and test set sequences, . **COV** is multiplied by 100% and **MMD** is multiplied by 10^2. For more results, please refer to the supplementary materials.

5.3. Generation Diversity

In this section we show the generation diversity of our proposed CAD-Diffuser. As many models on CAD sequence reconstruction can only generate *one* CAD construction sequence given the geometric input, we conduct an experiment on unconditional generation and compare our model with prevailing CAD generative models like [45, 48, 49]. In this part, we train a shape code generator like [49] and it randomly generates point cloud tokens. Following [45, 48, 49], we test the language diversity of generated CAD sequences via (1) distinct token n-grams, (2) summary statistics over pairwise DeepCAD embeddings and (3) summary statistics on pairwise sequence edit distances. We also evaluate the (4) *Coverage* (**COV**) percentage, (5) *Minimum Matching Distance* scores in the generated set of CAD sequences. These two scores can be regarded as geometric diversity measures since COV represents the proportion of reference set that matches the generated CAD sequence and MMD measures the Chamfer Distance between these matches. The results are listed in Table 2. The results show that the generated CAD sequences are far more diverse than exist-



Ground Truth	[SOL] Line 128 128 Line 223 128 Line 223 223 Line 128 223 Extrude 192 64 192 96 128 32 192 136 128 New_body One_sided [SOL] Circle 176 176 Extrude 192 64 192 96 128 224 192 128 64 New_body One_sided [EOS]
t = 2000	[M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] ·····[M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M] [M]
t = 1500	[SOL] [M] [M] [M] ·····[M] [M] [M] [M] [M] [M] Extrude [M] [M] [M] ·····[M] [M] Extrude [M] [M] [M] [M] ·····[M] [M] [M] [EOS]
t = 1000	[SOL] Line 128 [M] [M] [M] [M] Line [M] [M] Line [M] [M] Extrude [M] [M] 128 ····· Circle [M] [M] Extrude [M] [M] ····· [M] New_body[M] [EOS]
t = 750	[SOL] Line 128 [M] Line [M] [M] Line 223 223 Line 128 [M] Extrude [M] [M] [M] [M] 32 192 136 [M] New_body One_sided [SOL] Circle 176 [M] Extrude [M] [M] [M] [M] [M] [M] 224 [M] 128 64 New_body One_sided [EOS]
t = 500	[SOL] Line 128 128 Line 223 [M] Line 223 223 Line 128 223 Extrude 192 [M] [M] 96 128 32 192 136 [M] New_body One_sided [SOL] Circle 176 176 Extrude [M] [M] [M] 128 224 [M] 128 64 New_body One_sided [EOS]
t = 250	[SOL] Line 128 128 Line 223 128 Line 223 223 Line 128 223 Extrude 192 64 192 96 128 32 192 136 [M] New_body One_sided [SOL] Circle 176 176 Extrude [M] 64 192 96 128 224 192 128 64 New_body One_sided [EOS]
t = 0	[SOL] Line 128 128 Line 223 128 Line 223 223 Line 128 223 Extrude 192 64 192 96 128 32 192 136 128 New_body One_sided [SOL] Circle 176 176 Extrude 192 64 192 96 128 224 192 128 64 New_body One_sided [EOS]

Figure 4. Successive stages of denoising by our method where tokens represents the outlines are generated first owing to our volume-based noise schedule. [M] represents the mask tokens and ..... represnets all masked tokens for brevity.

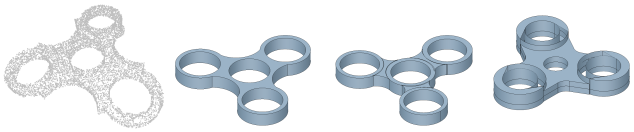


Figure 5. Qualitative example of reconstruction diversity given a single input point cloud. It can be seen that CAD-Diffuser strikes a balance between accuracy and diversity. It also provides designs of similar shape but with different styles, bringing new inspirations to designers.

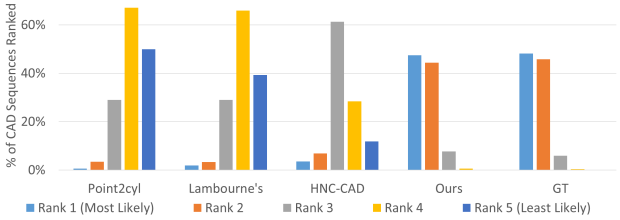


Figure 6. Distribution of votes on which of the reconstructed sequences are more likely to be reused by human designers. 7 human evaluators participate in the reconstructed results of all methods and the ground truth. Full results can be seen in the supplementary material.

ing methods in view of both CAD sequences and their rendered geometries, which is believed to be the effect of fine-grained control exerted by our proposed diffusion strategy. Figure 5 qualitatively showcases the generation diversity, which brings new shapes and inspirations to designers even under the point cloud condition is given. More results, some qualitative examples of unconditional generation, and the methods of generating Figure 5 are listed in the supplementary materials.

5.4. Visualization of Backward Process

In this section we study how CAD-Diffuser gradually unmasks the original CAD sequences. We select a CAD model and record its emergence at different timesteps, as shown in Figure 4. From the visualization we can observe that the outlines are generated first, namely the top-down design strategy is learned by our model, see Figure 3 for more illustrations.

5.5. User Study

To evaluate how much the reconstructed CAD sequence is like the human designed CAD sequences, we conduct a user study like [17]. For each CAD model, we shuffle and display the reconstructed CAD sequence and its corresponding *rendered geometry* of different reconstruction methods and mix them with the original CAD model. 7 users with design knowledge were asked to rank the likeliness of whether they are likely to reuse the CAD sequence on 50 reconstructions.

The result of human evaluation is displayed in Figure 6. From the Figure we can observe that our reconstruction is far more welcomed than reconstruction-based and search-based methods and even approaches the ground truth. This is because of the strong reconstruction performance of our method, as well as the outline-first noise schedule vividly imitates human design patterns in a soft manner during the training process. It is worth noting that HNC-CAD has achieves comparable performance, which demonstrates the importance of fine-grained generation by the side.

6. Conclusion

We introduce a novel multimodal token-based diffusion model for reconstruction CAD construction sequences from point clouds. The intuition of our approach is the unification of point clouds and CAD construction sequences at the token level. A random walk between tokens of each modality and [MASK] tokens blended the diffusion strategy into masked token modeling process of BERT. The proposed volume-based noise schedule vividly imitates the top-down design strategy of human designers. Experimental results demonstrate both the reconstruction fidelity and the generation diversity of our method. Future works include utilizing the pretrained models and large language models to build a more powerful reconstruction model for comprehensive CAD design human interface.

Acknowledgments. We thank the anonymous reviewers for their valuable suggestions. This work was supported by National Key Research and Development Program of China, No.2018YFB1402600.

References

[1] Opencascade. https://www.opencascade.com/. Accessed: 20-Oct-2023. 6, 7

[2] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021. 3, 5

[3] Arindam Banerjee, Inderjit S Dhillon, Joydeep Ghosh, Suvrit Sra, and Greg Ridgeway. Clustering on the unit hypersphere using von mises-fisher distributions. *Journal of Machine Learning Research*, 6(9), 2005. 6

[4] Dorit Borrmann, Jan Elseberg, Kai Lingemann, and Andreas Nüchter. The 3d hough transform for plane detection in point clouds: A review and a new accumulator design. *3D Research*, 2(2):1–13, 2011. 2

[5] Jiaao Chen, Aston Zhang, Mu Li, Alex Smola, and Diyi Yang. A cheaper and better diffusion language model with soft-masked noise. *arXiv preprint arXiv:2304.04746*, 2023. 3

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2, 4

[7] Paul Dierckx. An algorithm for fitting data over a circle using tensor product splines. *Journal of computational and applied mathematics*, 15(2):161–173, 1986. 2

[8] Richard O Duda and Peter E Hart. Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972. 2

[9] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 2

[10] Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and LingPeng Kong. Diffuseq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*, 2022. 2, 3, 4, 5

[11] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022. 3

[12] Haoxiang Guo, Shilin Liu, Hao Pan, Yang Liu, Xin Tong, and Baining Guo. Complexgen: Cad reconstruction by b-rep chain complex generation. *ACM Transactions on Graphics (TOG)*, 41(4):1–18, 2022. 2

[13] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 5

[14] Zhengfu He, Tianxiang Sun, Qiong Tang, Kuanning Wang, Xuanjing Huang, and Xipeng Qiu. DiffusionBERT: Improving generative masked language models with diffusion models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4521–4534, Toronto, Canada, 2023. Association for Computational Linguistics. 2, 3, 4, 5

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2, 4, 5

[16] Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. In *Advances in Neural Information Processing Systems*, pages 12454–12465. Curran Associates, Inc., 2021. 3

[17] Pradeep Kumar Jayaraman, Joseph G Lambourne, Nishkrit Desai, Karl DD Willis, Aditya Sanghi, and Nigel JW Morris. Solidgen: An autoregressive model for direct b-rep synthesis. *arXiv preprint arXiv:2203.13944*, 2022. 2, 8

[18] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019. 3, 4, 5

[19] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 6

[20] Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. Abc: A big cad model dataset for geometric deep learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9601–9611, 2019. 2

[21] Joseph George Lambourne, Karl Willis, Pradeep Kumar Jayaraman, Longfei Zhang, Aditya Sanghi, and Kamal Rahimi Malekshan. Reconstructing editable prismatic cad from rounded voxel models. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 2, 7

[22] Pu Li, Jianwei Guo, Xiaopeng Zhang, and Dong-Ming Yan. Secad-net: Self-supervised cad reconstruction by learning sketch-extrude operations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16826, 2023. 2, 7

[23] Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:4328–4343, 2022. 3

[24] Yifan Li, Kun Zhou, Wayne Xin Zhao, and Ji-Rong Wen. Diffusion models for non-autoregressive text generation: A survey. *arXiv preprint arXiv:2303.06574*, 2023. 2, 4, 5

[25] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 5

[26] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 6

[27] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 6

[28] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2, 3

[29] Daxuan Ren, Jianmin Zheng, Jianfei Cai, Jiatong Li, and Junzhe Zhang. Extrudenet: Unsupervised inverse sketch-and-extrude for shape parsing. In *European Conference on Computer Vision*, pages 482–498. Springer, 2022. 2, 7

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3, 4

[31] Ruwen Schnabel, Roland Wahl, and Reinhard Klein. Efficient ransac for point-cloud shape detection. In *Computer graphics forum*, pages 214–226. Wiley Online Library, 2007. 2

[32] Gopal Sharma, Difan Liu, Subhransu Maji, Evangelos Kalogerakis, Siddhartha Chaudhuri, and Radomír Měch. Parsenet: A parametric surface fitting network for 3d point clouds. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 261–276. Springer, 2020. 2

[33] Dmitriy Smirnov, Mikhail Bessmeltsev, and Justin Solomon. Learning manifold patch-based representations of man-made shapes. *arXiv preprint arXiv:1906.12337*, 2019. 2

[34] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 2, 4, 5

[35] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[36] C-H Teh and Roland T. Chin. On the detection of dominant points on digital curves. *IEEE Transactions on pattern analysis and machine intelligence*, 11(8):859–872, 1989. 2

[37] Mikaela Angelina Uy, Yen-Yu Chang, Minhyuk Sung, Purvi Goel, Joseph G Lambourne, Tolga Birdal, and Leonidas J Guibas. Point2cyl: Reverse engineering 3d objects from point clouds to extrusion cylinders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11850–11860, 2022. 2, 7

[38] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 3

[39] Alex Wang and Kyunghyun Cho. Bert has a mouth, and it must speak: Bert as a markov random field language model. *arXiv preprint arXiv:1902.04094*, 2019. 3

[40] Kehan Wang, Jia Zheng, and Zihan Zhou. Neural face identification in a 2d wireframe projection of a manifold object. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1622–1631, 2022. 2

[41] Xiaogang Wang, Yuelang Xu, Kai Xu, Andrea Tagliasacchi, Bin Zhou, Ali Mahdavi-Amiri, and Hao Zhang. Pie-net: Parametric inference of point cloud edges. *Advances in neural information processing systems*, 33:20167–20178, 2020. 2

[42] Alexander Wettig, Tianyu Gao, Zexuan Zhong, and Danqi Chen. Should you mask 15% in masked language modeling? *arXiv preprint arXiv:2202.08005*, 2022. 5

[43] Karl DD Willis, Pradeep Kumar Jayaraman, Joseph G Lambourne, Hang Chu, and Yewen Pu. Engineering sketch generation for computer-aided design. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2105–2114, 2021. 2

[44] Karl DD Willis, Yewen Pu, Jieliang Luo, Hang Chu, Tao Du, Joseph G Lambourne, Armando Solar-Lezama, and Wojciech Matusik. Fusion 360 gallery: A dataset and environment for programmatic cad construction from human design sequences. *ACM Transactions on Graphics (TOG)*, 40(4): 1–24, 2021. 6

[45] Rundi Wu, Chang Xiao, and Changxi Zheng. Deepcad: A deep generative network for computer-aided design models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6772–6782, 2021. 2, 5, 6, 7

[46] Yu Xiang, Christopher Xie, Arsalan Mousavian, and Dieter Fox. Learning rgb-d feature embeddings for unseen object instance segmentation. In *Conference on Robot Learning*, pages 461–470. PMLR, 2021. 6

[47] Xianghao Xu, Wenzhe Peng, Chin-Yi Cheng, Karl DD Willis, and Daniel Ritchie. Inferring cad modeling sequences using zone graphs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6062–6070, 2021. 2

[48] Xiang Xu, Karl DD Willis, Joseph G Lambourne, Chin-Yi Cheng, Pradeep Kumar Jayaraman, and Yasutaka Furukawa. Skexgen: Autoregressive generation of cad construction sequences with disentangled codebooks. In *International Conference on Machine Learning*, pages 24698–24724. PMLR, 2022. 7

[49] Xiang Xu, Pradeep Kumar Jayaraman, Joseph G Lambourne, Karl DD Willis, and Yasutaka Furukawa. Hierarchical neural coding for controllable cad model generation. *arXiv preprint arXiv:2307.00149*, 2023. 2, 4, 6, 7

[50] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19313–19322, 2022. 6

[51] Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Fei Huang, and Songfang Huang. Seqdiffuseq: Text diffusion with encoder-decoder transformers. *ArXiv*, abs/2212.10325, 2022. 2

[52] Junyi Zhang, Jiaqi Guo, Shizhao Sun, Jian-Guang Lou, and Dongmei Zhang. Layoutdiffusion: Improving graphic layout generation by discrete diffusion probabilistic models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7226–7236, 2023. 3

[53] Kun Zhou, Yifan Li, Wayne Xin Zhao, and Ji-Rong Wen. Diffusion-nat: Self-prompting discrete diffusion for non-autoregressive text generation. *arXiv preprint arXiv:2305.04044*, 2023. 2, 3

[54] Hao Zou, Zae Myung Kim, and Dongyeop Kang. Diffusion models in nlp: A survey. *arXiv preprint arXiv:2305.14671*, 2023. 2