

# Diffusion-SDF: Text-to-Shape via Voxelized Diffusion

Muheng Li<sup>1</sup>, Yueqi Duan<sup>†,2</sup>, Jie Zhou<sup>1</sup>, Jiwen Lu<sup>1</sup>

<sup>1</sup>Department of Automation, Tsinghua University

<sup>2</sup>Department of Electronic Engineering, Tsinghua University

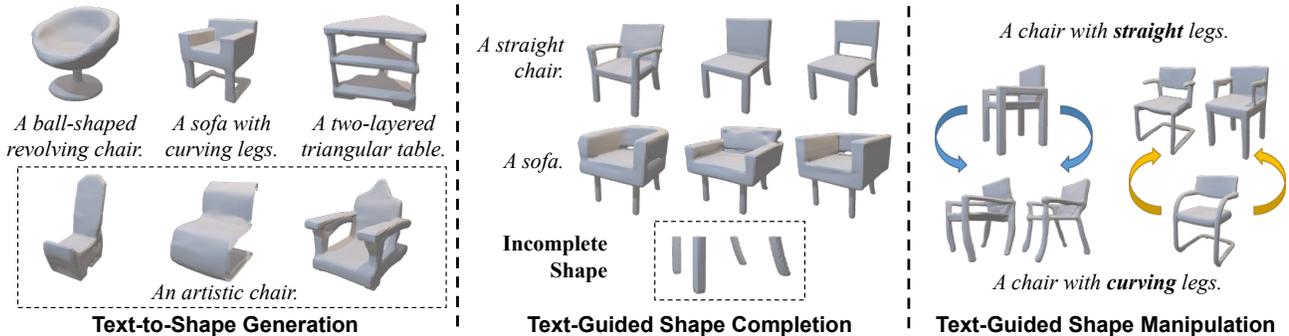


Figure 1. We propose a text-to-shape synthesis approach named Diffusion-SDF. Our method is capable of performing various text-to-shape tasks including text-to-shape generation (directly generating 3D shapes from texts), text-guided shape completion (generating the missing part of an incomplete shape under the guidance of texts), and text-guided shape manipulation (modifying a given shape according to texts).

## Abstract

With the rising industrial attention to 3D virtual modeling technology, generating novel 3D content based on specified conditions (e.g. text) has become a hot issue. In this paper, we propose a new generative 3D modeling framework called Diffusion-SDF for the challenging task of text-to-shape synthesis. Previous approaches lack flexibility in both 3D data representation and shape generation, thereby failing to generate highly diversified 3D shapes conforming to the given text descriptions. To address this, we propose a SDF autoencoder together with the Voxelized Diffusion model to learn and generate representations for voxelized signed distance fields (SDFs) of 3D shapes. Specifically, we design a novel UinU-Net architecture that implants a local-focused inner network inside the standard U-Net architecture, which enables better reconstruction of patch-independent SDF representations. We extend our approach to further text-to-shape tasks including text-conditioned shape completion and manipulation. Experimental results show that Diffusion-SDF is capable of generating both high-quality and highly diversified 3D shapes that conform well to the given text descriptions. Diffusion-SDF has demonstrated its superiority compared to previous state-of-the-art text-to-shape approaches.

<sup>†</sup>Corresponding author.

## 1. Introduction

Exploring data representations for 3D shapes has been a fundamental and critical issue in 3D computer vision. Explicit 3D representations including point clouds [38, 39], polygon meshes [17, 24] and occupancy voxel grids [9, 53] have been widely applied in various 3D downstream applications [1, 35, 56]. While explicit 3D representations achieve encouraging performance, there are some primary limitations including not being suitable for generating watertight surfaces (e.g. point clouds), or being subject to topological constraints (e.g. meshes). On the other hand, implicit 3D representations have been widely studied more recently [3, 14, 37], where representative works including DeepSDF [37], Occupancy Network [32] and IM-Net [8]. In general, implicit functions encode the shapes by the iso-surface of the function, which is a continuous field and can be evaluated at arbitrary resolution.

In recent years, numerous explorations have been conducted for implicit 3D generative models, which show promising performance on several downstream applications such as single/multi-view 3D reconstruction [23, 54] and shape completion [12, 34]. Besides, several studies have also explored the feasibility of directly generating novel 3D shapes based on implicit representations [15, 21]. However, these approaches are incapable of generating specified 3D shapes that match a given condition, e.g. a short text describing the shape characteristics as shown in Figure 1.

Text-based visual content synthesis has the advantages of the flexibility and generality [41, 42]. Users may generate rich and diverse 3D shapes based on easily obtained natural language descriptors. In addition to generating 3D shapes directly based on text descriptions, manipulating 3D data with text guidance can be further utilized for iterative 3D synthesis and fine-grained 3D editing, which can be beneficial for non-expert users to create 3D visual content.

In the literature, there have been few attempts on the challenging task of text-to-shape generation based on implicit 3D representations [29, 34, 48]. For example, AutoSDF [34] introduced a vector quantized SDF autoencoder together with an autoregressive generator for shape generation. While encouraging progress has been made, the quality and diversity of generated shapes still requires improvement. The current approaches struggle to generate highly diversified 3D shapes that both guarantee generation quality and conform to the semantics of the input text. Motivated by the success of denoising diffusion models in 2D image [13, 19, 36] and even explicit 3D point cloud [31, 58, 59] generation, we find that DMs achieve high-quality and highly diversified generation while being robust to model training. To this end, we aim to design an implicit 3D representation-based generative diffusion process for text-to-shape synthesis that can achieve better generation flexibility and generalization performance.

In this paper, we propose the *Diffusion-SDF* framework for text-to-shape synthesis based on truncated signed distance fields (TSDFs). Considering that 3D shapes share structural similarities at local scales, and the cubic data volume of 3D voxels may lead to slow sampling speed for diffusion models, we propose a two-stage separated generation pipeline. First, we introduce a patch-based SDF autoencoder that map the original signed distance fields into patch-independent local Gaussian latent representations. Second, we introduce the *Voxelized Diffusion* model that captures the intra-patch information along with both patch-to-patch and patch-to-global relations. Specifically, we design a novel *UinU*-Net architecture to replace the standard U-Net [46] for the noise estimator in the reverse process. *UinU*-Net implants a local-focused inner network inside the outer U-Net backbone, which takes into account the patch-independent prior of SDF representations to better reconstruct local patch features from noise. Our work digs deeper into the further potential of diffusion model-based approaches towards text-conditioned 3D shape synthesis based on voxelized TSDFs. Extensive experiments on the largest existing text-shape dataset [6] illustrate that our *Diffusion-SDF* approach has achieved promising generation performance on several text-to-shape tasks with regard to the existing state-of-the-art approaches in both qualitative

and quantitative evaluative dimension.

## 2. Related Work

**Text-conditioned generative 3D models.** In the field of generative 3D modeling, a variety of works have focused on synthesizing 3D visual content unconditionally [1, 32, 52, 56] or conditioned on single/multi-view images [9, 34, 54, 55]. Besides, there has also been a series of works concentrating on the challenging task of text-to-shape generation [6]. Some of them adopted purely explicit 3D data representation-based methods to generate 3D shapes conditioned on input text [6, 33, 58]. In contrast, our focus lies on synthesizing 3D shapes based on implicit 3D data representations. So far, there have also been few existing works focusing on the task of implicit text-to-shape generation [29, 34, 48]. All of these approaches have yielded impressive generation results, but there are still some remained issues to be addressed. Sanghi *et al.* [48] proposed a normalizing flow [43]-based approach to generate shape voxels using implicit Occupancy Networks [32], and Liu *et al.* [29] introduced a shape-IMLE [28] generator using implicit IMNET [8] decoder, while these approaches neither employ the more flexible implicit SDFs as data representation nor take into account the constraints for local shape structures. More similar to our work, AutoSDF [34] introduced an autoregressive prior for 3D shape generation based on a discretized SDF autoencoder. The autoregressive model adopts a relatively unnatural way to predict patched 3D tokens in a sequential manner that loses the 3D-specific inductive bias and is also relatively inefficient.

**Diffusion Probabilistic Models.** Diffusion Probabilistic Models (DPMs) [19, 49], also known as *diffusion models*, have currently arisen as a powerful family of generative models. Compared to previous state-of-the-art generative models including Generative Adversarial Network (GAN) [16], Variational Autoencoder (VAE) [26], and Flow-based generative models [43], diffusion model shows its superiority in both training stability and generative diversity [11]. Diffusion models have achieved promising performance on image [13, 19, 36, 45] and speech [7, 27] synthesis. Especially, fairly impressive results have been achieved with diffusion model-based approaches on the task of text-to-image synthesis [41, 45, 47]. In the field of 3D computer vision, several studies have adopted diffusion models for generative 3D modeling [31, 58, 59]. PVD [59] employed diffusion models to generate 3D shapes based on point-voxel 3D representation. Luo *et al.* [31] treated points in point clouds as particles in a thermodynamic system with a heat bath. LION [58] introduced a VAE framework with hierarchical diffusion models in latent space. All these approaches have focused on the diffusion process of explicit 3D data representations for shape generation. On the contrary, our work tries to explore the feasibility of diffusion

---

Strictly, our approach utilizes a combined explicit-implicit representation as voxelized signed distance fields.

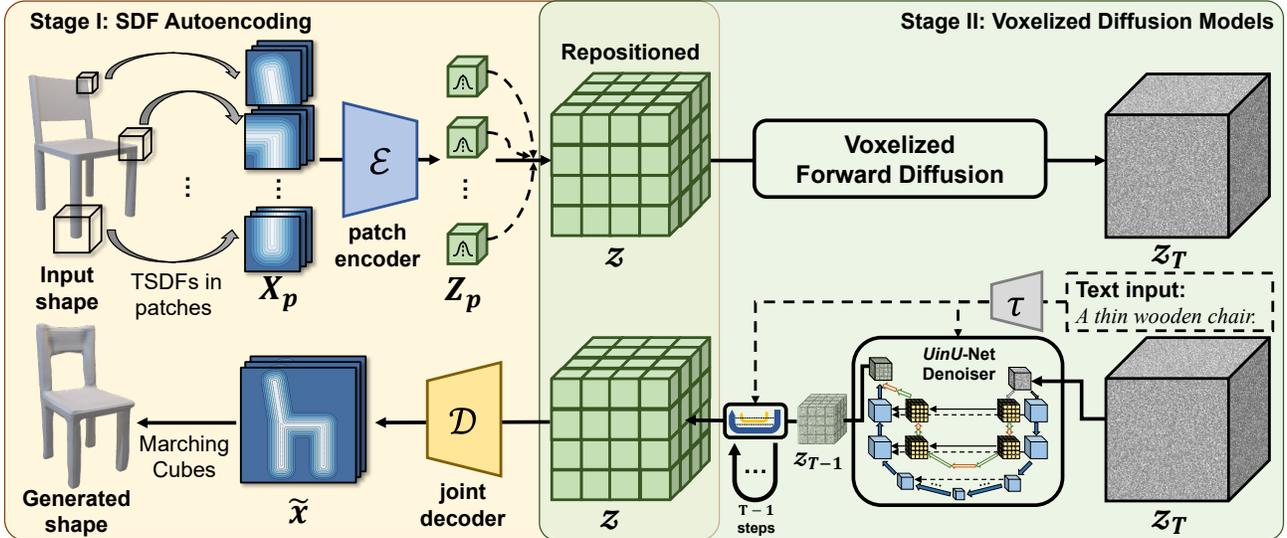


Figure 2. **Overview of Diffusion-SDF.** We propose a two-stage pipeline to generate novel 3D shapes from texts. First, we train an autoencoder to learn patch-independent normal-distributed representations for voxelized TSDFs. Then, we propose the Voxelized Diffusion framework together with the *UinU-Net* denoiser to generate shape representations. Please check Figure 3 for more details of *UinU-Net*.

models on implicit 3D data representations.

### 3. Method

In this section, we introduce the methodology design for *Diffusion-SDF*. In general, we propose a two-stage pipeline as illustrated in Figure 2. Detailed information of each stage will be included in the following sections.

#### 3.1. Autoencoding Signed Distance Fields

Signed distance fields (SDF) belong to a type of implicit 3D data representation which assigns the scalar signed distance value to the shape surface for each point  $p$  in the 3D space  $\mathbb{R}^3$ . Our objective is to generate the signed distance fields for the target 3D shapes that match the given text conditions. Specifically, the 3D data are represented by truncated signed distance functions (TSDF) in a regularly-spaced voxel grid as [3, 22]. Generating voxelized SDFs directly from denoising diffusion models is both cost-expensive and time-consuming. In addition, the local structural information of 3D shapes is not well-emphasized through direct voxel-based generation. To address this, we propose a patch-wise autoencoder to learn the latent representations for voxelized signed distance fields.

Given a 3D shape, we first sample its truncated signed distance field  $x$  as a voxel grid of size  $D^3$ . Before the shape  $x$  is encoded, it is first split into  $N$  local patches of size  $P^3$ , producing a sequence of shape patches  $X_p = [x_{p1}, \dots, x_{pN}]$ , where  $N = (D/P)^3$  is the resulting number of patches. Then, the local shape encoder  $\mathcal{E}_{loc}$  encodes each shape patches into latent representations  $Z_p = [z_{p1}, \dots, z_{pN}]$ , where  $z_{pn} = \mathcal{E}_{loc}(x_{pn}) \in \mathbb{R}^{d \times d \times d \times c}$  and  $c$  is

the number of latent channels. Here, since each patch is encoded independently, the local structural information can be explored well through the local shape encoder. Meanwhile, the input data scale can also be downsampled by the factor  $P$ . However, when reconstructing the shape from local patches, it is also necessary to consider the spatial location of each patch in the global shape, and the interrelationship between adjacent patches. To preserve both patch-to-global and patch-to-patch information while decoding the latent embeddings, the patch embeddings are first rearranged into a voxel grid embedding  $z$ , and then sent into the patch-joint decoder  $\mathcal{D}$  that reconstructs the SDF field from the latent patches, giving  $\tilde{x} = \mathcal{D}(z)$ .

In detail, we adopt a VAE [26, 44]-like autoencoder, which encodes each shape patch into a normal distribution. We utilize a combination of both  $\mathcal{L}_1$  reconstruction loss together with the KL-regularization loss at the training stage for the SDF autoencoder. The latter one forces a mild KL-penalty towards a standard normal distribution on each learned patch latent. The final SDF representations will be in the form of patch-independent Gaussian distributions.

#### 3.2. Voxelized Diffusion Models (VDMs)

**Diffusion models (DMs)** [19, 49] are a class of probabilistic generative models that learn to fit a certain distribution by gradually denoising a Gaussian variable through a fixed Markov Chain of length  $T$ . Given a data sample  $x_0 \sim q(x_0)$ , DMs describe two different processes in the opposite direction: a *forward process*  $q(x_{0:T})$  that gradually transform a data sample into pure Gaussian noise, and a *reverse process*  $p_\theta(x_{0:T})$  that gradually denoise a pure

Gaussian sample into real data,

$$\begin{aligned} q(\mathbf{x}_{0:T}) &= q(\mathbf{x}_0) \prod_{t=1}^T q(\mathbf{x}_t | \mathbf{x}_{t-1}), \\ p_\theta(\mathbf{x}_{0:T}) &= p(\mathbf{x}_T) \prod_{t=1}^T p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t), \end{aligned} \quad (1)$$

where  $q(\mathbf{x}_t | \mathbf{x}_{t-1})$  and  $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$  are both Gaussian transition probabilities in the forms of

$$\begin{aligned} q(\mathbf{x}_t | \mathbf{x}_{t-1}) &= \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t} \mathbf{x}_{t-1}, \beta_t \mathbf{I}), \\ p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) &= \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \beta_t \mathbf{I}). \end{aligned} \quad (2)$$

Based on [19], the mean variable  $\boldsymbol{\mu}_\theta(\mathbf{x}_t, t)$  for the reverse transition  $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$  can be expressed in the form of

$$\boldsymbol{\mu}_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}} \left( \mathbf{x}_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right), \quad (3)$$

where  $\alpha_t = 1 - \beta_t$ ,  $\bar{\alpha}_t = \prod_{i=1}^t \alpha_i$ , and  $\beta_t$  decreases to 0 gradually as  $t$  approaches 0. The evidence lower bound (ELBO) is maximized at the training stage of DMs, eventually leading to the loss function as the form of

$$\mathcal{L}_{DM} = \mathbb{E}_{\mathbf{x}, t, \boldsymbol{\epsilon} \sim \mathcal{N}(0,1)} \left[ \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|^2 \right], \quad (4)$$

where  $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \boldsymbol{\epsilon}$ , and  $t$  is uniformly sampled from  $\{1, \dots, T\}$ . The neural network-based score estimator  $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$  is the core function of denoising diffusion models, which is implemented as a timestep-conditioned denoising autoencoder in [19].

**UinU-Net Architecture.** In our scenario, we are designing a novel diffusion-based architecture to generate latent SDF representations with regard to the target text conditions. The corresponding TSDF fields can be then reconstructed through the patch-joint decoder which is pre-trained in the first stage. As we mentioned in the above section, the key part of generative diffusion models is to estimate the function approximator  $\boldsymbol{\epsilon}_\theta$  from the data distribution of the training set. Since our model is designed to generate latent samples, the score estimator can be expressed as  $\boldsymbol{\epsilon}_\theta(z_t, t)$ , as the loss function becomes

$$\mathcal{L}_{Diffusion-SDF} = \mathbb{E}_{z, t, \boldsymbol{\epsilon} \sim \mathcal{N}(0,1)} \left[ \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(z_t, t)\|^2 \right]. \quad (5)$$

To incorporate the 3D positional prior of different latent shape patches, we adopt a 3D U-Net [10]-based autoencoder as the neural function approximator  $\boldsymbol{\epsilon}_\theta$  that directly learns to denoise the voxel grid embedding  $z_t$  that is obtained from the pre-trained local shape encoder  $\mathcal{E}_{loc}$ . Besides, another crucial prior for our design is the patch-based encoding process, through which all the shape patches are encoded into independent Gaussian distributions. Thus, when generating the local shape embeddings, an important point is to recover the independent distribution for

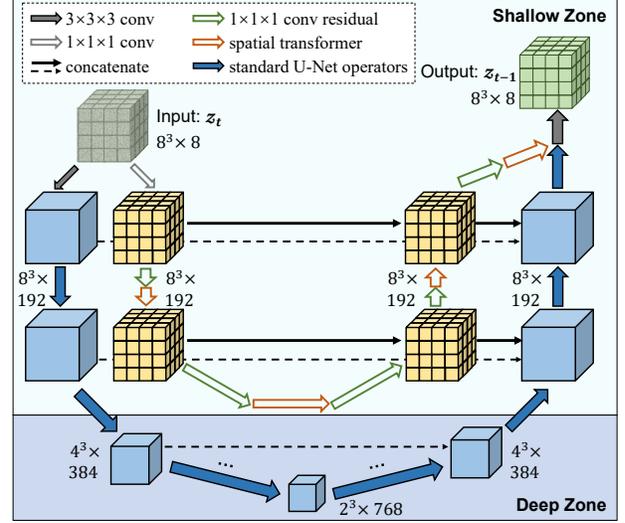


Figure 3. **Detailed illustration of UinU-Net.** To better recover  $z$  based on independently distributed patches, we propose to implant an inner network inside the standard U-Net architecture. The inner network is mainly composed of  $1 \times 1 \times 1$  convolutions, thus it focuses on patch-wise features, while the spatial transformer brings in patch-to-patch information. (Standard U-Net operators include  $3 \times 3 \times 3$  residual blocks, conditional cross-attention layers, pooling layers and up-scaling layers.)

each shape patch as well. To address this, we propose a novel UinU-Net architecture for the implementation of the autoencoder-based neural score estimator as shown in Figure 3. The standard 3D U-net [10] adopts a series of  $3 \times 3 \times 3$  convolutional layers to construct a downsampling-upsampling network architecture with the information shared through skip connections. This design takes into account the patch-to-patch and patch-to-global features via hierarchical receptive fields. Given that our input data are compressed 3D latent representations, and each patch is encoded independently, we propose to implant another inner network inside the outer U-Net architecture where the embedded resolution is equal to the original latent resolution  $D/P$ . Specifically, we adopt a  $1 \times 1 \times 1$  convolution-based ResNet [18] structure to learn independent patch-focused information. We also introduce the spatial Transformer [51] network that accepts positional embedded patch representations as input tokens, following self-attention layers to capture the relational information between independent local patch embeddings. The inner synthesis paths are skip-connected to the outer ones, ensuring information transmission from inner to outer networks. This architecture is designed to capture the intra-patch information along with both patch-to-patch and patch-to-global relations.

**Text-Guided Shape Generation** So far, we have discussed the generative process without text conditions. To synthe-

size latent SDF representations based on given conditions, we introduce a conditioning mechanism based on classifier-free guidance diffusion [20]. We adopt a text-conditioned score estimator. In detail, to fit the target conditions into the 3D denoising autoencoder, we utilize the cross-attention mechanism as proposed in [45]. The input caption is first encoded as text embeddings through a pre-trained text encoder  $\tau$ . Thus, the conditioned score can be formed as  $\epsilon_\theta(z_t|\tau(c))$ , where  $c$  is the input caption. Meanwhile, we also obtain an unconditioned score  $\epsilon_\theta(z_t|\emptyset)$  with an empty sequence as input caption. Within the guided diffusion sampling process, the output of the model is extrapolated further in the direction of  $\epsilon_\theta(z_t|\tau(c))$  and away from  $\epsilon_\theta(z_t|\emptyset)$  as:

$$\hat{\epsilon}_\theta(z_t|\tau(c)) = \epsilon_\theta(z_t|\emptyset) + s \cdot (\epsilon_\theta(z_t|\tau(c)) - \epsilon_\theta(z_t|\emptyset)), \quad (6)$$

where  $s \geq 1$  refers to the guidance scale.

**Text-Guided Shape Completion** Current shape completion approaches mainly focused on recovering the full shapes from partial input shapes or single view images [1, 57, 59]. Besides, we have explored the feasibility of text-conditioned shape completion. According to the patch-based design of our SDF encoder, information for the given shape patches can be encoded even if some of the patches are missing. Motivated by the success of diffusion model-based image inpainting [30, 45], we introduce a mask-diffusion strategy to generate the missing shape based on a pre-trained VDM. The core idea is to incorporate known information into each denoising step of the reverse process. For a full shape representation, we transform it into a partial shape representation given the mask  $m_z$ . For each estimation step  $z_{t-1}$ , we combine the estimated result through the reverse process  $\tilde{z}_{t-1}$  with the forward sampling result of the unmasked patches  $\hat{z}_{t-1}$ ,

$$z_{t-1} = (1 - m) \odot \tilde{z}_{t-1} + m \odot \hat{z}_{t-1}, \quad (7)$$

where  $\odot$  refers to element-wise product, and

$$\begin{aligned} \tilde{z}_{t-1} &\sim \mathcal{N}(\mu_\theta(z_t, t, c), \beta_t \mathbf{I}) \\ \hat{z}_{t-1} &= \sqrt{\bar{\alpha}_t} z_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon. \end{aligned} \quad (8)$$

By such a strategy, the unknown region can be recovered based on both the given shape and text conditions.

**Text-Guided Shape Manipulation** Visual content editing is one of the most valuable applications for vision-language generation. We propose a diffusion-based text-guided shape manipulation approach. For a given shape representation  $z_{init}$ , our goal is to transform it into another shape representation  $z_{goal}$  based on text  $c$ . Inspired by image manipulation techniques [4, 25], we use a cycle-sampling strategy based on a pre-trained VDM. Firstly, we forward-sample the given shape  $z_{init}$  for  $t_{mid}$  steps, where  $0 < t_{mid} < T$ . This operation will lead to an intermediate output  $z_{mid}$

$$z_{mid} = \sqrt{\bar{\alpha}_{t_{mid}}} z_{init} + \sqrt{1 - \bar{\alpha}_{t_{mid}}} \epsilon. \quad (9)$$

Then, the reverse process will start from  $z_{mid}$ , and another  $t_{mid}$  denoising steps will be conducted conditioned on  $c$ . Such a design will make the generated shapes correspond to the target text descriptions while maintaining the original shape characteristics.

## 4. Experiments

To evaluate the effectiveness of our proposed *Diffusion-SDF*, we conduct extensive experiments on several text-conditioned 3D shape synthesis tasks including text-to-shape generation, text-conditioned shape completion, and text-guided shape manipulation. The following subsections describe the details of the experimental settings, results, and analyses.

**Dataset.** We mainly trained and evaluated our approach on the current largest text-shape dataset Text2Shape (T2S) [6]. T2S gathered data in the form of shape-text pairs based on two object classes (chairs, tables) in ShapeNet [5]. T2S contains about 75K shape-text pairs in total ( $\sim 30K$  for chairs,  $\sim 40K$  for tables), with an average of  $\sim 16$  words per description. T2S was originally designed to contain both color and shape information within the text descriptions, while we mainly focus on text-shape correspondence.

**Implementation details.** For the training stage of the SDF autoencoder, to improve the model’s generalization ability, we trained the autoencoder across the 13 categories of ShapeNet [5] dataset. For the input data, we sampled the voxelized SDF from original shapes with  $64 \times 64 \times 64$  grid points, which is also the data size adopted for follow-up operations. For the diffusion stage, to facilitate the generative sampling speed, we adopt the DDIM [50] sampler to reduce the original DDPM sampling steps from  $T = 1000$  to 50. For the conditioning mechanism, we adopt a freeze CLIP [40] text embedder as text encoder  $\tau$ .

### 4.1. Text-Conditioned Shape Generation

The most direct application for our approach is generating novel shapes conditioned on the given text descriptions. To demonstrate the effectiveness of our approach, we compare our generation results with two state-of-the-art supervised text-to-shape synthesis methods based on implicit 3D representations [29, 34]. Besides, there are some other works regarding text-to-shape generation, which have different settings compared to our approach. For instance, [48] proposed a zero-shot shape generation approach conditioned on categorical texts, and [15, 33] proposed to generate textured shapes based on provided/pre-generated meshes. Thus, these works are not compared directly to our approach.

**Quantitative comparison.** To compare the generation performance of our approach to the previous methods quantitatively, we propose to use several metrics to evaluate the generated results:



Figure 4. **Qualitative results of text-to-shape generation.** Our approach can generate shapes that conform to diverse text descriptions.

- **IoU** (Intersection over Union) measures the occupancy similarity between the generated shape to the ground truth. To compute the IoU score, we downsample all the generated SDFs to occupancy voxel grids of size  $32^3$ . IoU metric is used to measure the conformity of the generated samples with the ground truth shapes.
- **Acc** (Classification Accuracy) introduces a voxel-based classifier, which is pre-trained to classify the 13 categories in ShapeNet [5] according to [48]. Acc metric is used to measure the semantic authenticity of the generated samples.
- **CLIP-S** (CLIP Similarity Score) introduces the pre-trained vision-language model CLIP [40] for further evaluation. The CLIP can be used to measure the correspondence between given images and texts. We render 5 view images for each generated shape, and compute the cosine similarity score between rendered images and the given text. The highest score is reported for each text query. CLIP-S metric is used to measure the semantic conformance between the generated samples and input texts.
- **TMD** (Total Mutual Difference). For each given text description, we generate  $k = 10$  different samples. Then, we compute the average IoU score for each generated shape to other  $k - 1$  shapes. The average IoU score for all text queries is reported. TMD assesses the generation diversity for each given text query.

To keep consistency with previous approaches, we limit the comparisons to the chair category in Text2Shape [6] dataset. Since the officially released AutoSDF [34] model was trained on another dataset [2], we re-trained the model following the original settings on the training set of

Table 1. Quantitative comparisons of text-to-shape generation.

Methods	IoU $\uparrow$	Acc $\uparrow$	CLIP-S $\uparrow$	TMD $\downarrow$
Liu <i>et al.</i> [29]	0.160	34.79	29.94	0.891
AutoSDF [34]	0.187	83.88	29.10	0.581
Diffusion-SDF (Ours)	<b>0.194</b>	<b>88.56</b>	<b>30.88</b>	<b>0.169</b>

Text2Shape for the fair comparison. The results are shown in Table 1. From the results, it can be seen that our approach is able to achieve relatively high generation quality with high IoU and Acc scores, and robust text-shape conformance with good CLIP-S performance, while achieving much better generation diversity with a much lower TMD performance.

**Qualitative results.** The quantitative experiment shows the generation performance of our approach on the limited given subset from Text2Shape dataset. Due to the generalization of natural language, our approach is capable of generating different forms of shapes based on various text descriptions. Figure 4 shows the generated results of *Diffusion-SDF* from different text inputs. The results show that our method can generate eligible results from distinguished conditions. Besides, our method is capable of synthesizing various shape outputs from the same input text description. Figure 5 compares our approach with [29, 34] based on same text queries. The results have shown that our method has a great advantage in the diversity of generated samples, while the previous approaches are unable to generate highly-diversified shapes.

## 4.2. Ablation Studies

We have conducted extensive ablation studies to demonstrate the effectiveness of our special architecture design for *Diffusion-SDF*.

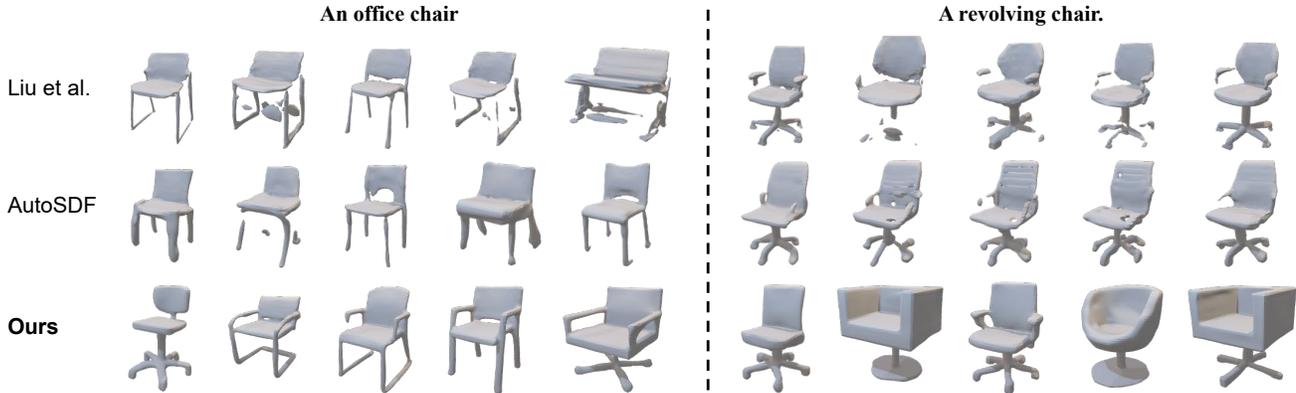


Figure 5. **Qualitative comparison of diversified text-to-shape generation.** Our approach is capable of generating highly-diversified shapes that conform to the same text query, while also keeping the satisfactory quality of the generated shapes.



Figure 6. **Qualitative comparison of U-Net and UinU-Net.** Since UinU-Net-based architecture contains a local-focused inner network, it can better recover the independent patch representations, thereby improving the generation quality compared to conventional U-Net.

**Effectiveness of the UinU-Net architecture.** In order to validate the benefit of our proposed modifications to the conventional U-Net architecture, we additionally compare our proposed UinU-Net architecture to the simpler U-Net architecture used in [45]. The quantitative results for text-to-shape generation are shown in Table 2. Qualitative comparisons are shown in Figure 6. From the results, it can be found that our proposed UinU-Net architecture for Voxelized Diffusion models generally improves the quality of generated shapes. With the UinU-Net architecture, the noise for each step is estimated both patch-mutually and patch-independently. Since the shape representations are repositioned based on separate patch embeddings extracted by the SDF encoders, the key to recovering the authentic shape SDFs from the joint decoder is to ensure that the generated SDF representations are also distributed independently in patches. From the qualitative results, it can be inferred that the inner network is conducive to recovering the independently distributed patch representations.

**Inner network architecture.** As for the UinU-Net architecture, we have conducted ablation experiments on several variants for the inner network to validate the effectiveness of the final design. The results have shown in Table 3. According to the results, it can be inferred that the inner outer concatenation mechanism and spatial attention layer can generally improve the generation quality. Without the inner-outer

Table 2. Evaluating effectiveness of the UinU-Net architecture.

	IoU $\uparrow$	Acc $\uparrow$	CLIP-S $\uparrow$	TMD $\downarrow$
U-Net [45]	0.187	80.98	29.35	0.171
UinU-Net (Ours)	<b>0.194</b>	<b>88.56</b>	<b>30.88</b>	<b>0.169</b>

Table 3. Evaluating effectiveness of the inner network design.

	IoU $\uparrow$	Acc $\uparrow$	CLIP-S $\uparrow$	TMD $\downarrow$
UinU-Net	0.194	<b>88.56</b>	<b>30.88</b>	0.169
- w/o in-out concat	<b>0.198</b>	83.91	28.68	0.181
- w/o spatial attention	0.194	85.56	30.82	<b>0.161</b>

concatenation mechanism, the accuracy score and CLIP-S have significant decreases, indicating that the information transmission mechanism has affected the semantic authenticity and conformance of the generated shapes. Accuracy score has dropped without the spatial attention module, indicating that the introduction of patch-to-patch information helps preserve the full shape semantics while recovering the patch-wise representations.

### 4.3. Text-Guided Shape Completion

In this section, we validate the effectiveness of our approach on text-guided shape completion task. Given a partial input shape, our goal is to generate the missing part from the partial shape conditioned on the input text descriptions.

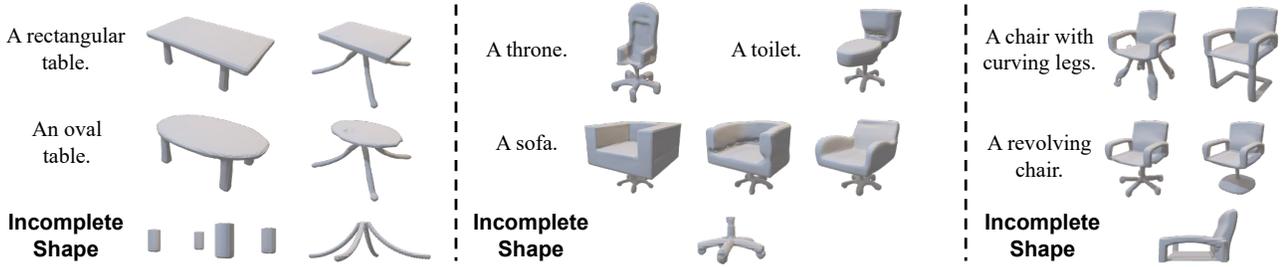


Figure 7. **Qualitative result of text-guide shape completion.** Given a known partial shape, our approach can generate the missing part based on text conditions.

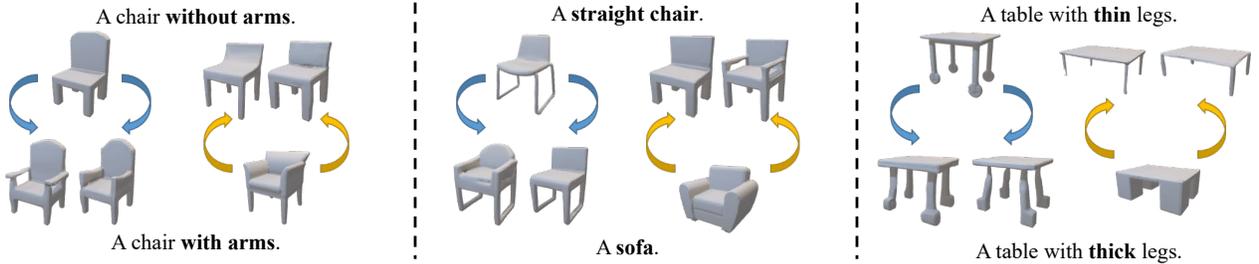


Figure 8. **Qualitative result of text-guide shape manipulation.** Given a known shape, our approach is able to manipulate the given shape into the target shape based on the text descriptions.

For example, our approach can generate the body of a chair based on the given chair legs, or generate the missing chair legs according to the body of the chair. The qualitative generation results are displayed in Figure 7. From the results, it can be seen that our approach is able to generate the missing part of a shape based on the conditioned text, while at the same time being well-blended with the given shape. Moreover, from the generated examples including *revolving throne*, *revolving toilet*, and *revolving sofa*, it can be inferred that our completion performance is not constrained by the shape geometry in the training set.

#### 4.4. Text-Guided Shape Manipulation

In this section, we demonstrate the effectiveness of our approach for text-guided shape manipulation task. Given a known shape and an instructive text description, our approach is capable of altering the given shape to the eligible style. Figure 8 gives several qualitative illustrations for our proposed approach. From the results, it can be summarized that our approach can handle the shape manipulation of both local shape structures (e.g. the existence/style of arms or legs) or the overall shape characteristics (e.g. sofa or straight chair). In addition, one key issue for our designed approach is the choice for  $t_{mid}$ . Since the shape characteristics usually form in the early stage of the reverse process, thus we set  $t_{mid}$  to relatively larger step numbers (600~800 step). In our experiments, we find that the original shape keeps unchanged when  $t_{mid}$  is too small, while overlarge  $t_{mid}$  also leads to the destruction of the original shape features.

## 5. Limitations and Conclusion

**Limitations.** In this paper, we have demonstrated the superiority of our approach on the Text2Shape [6] dataset. Nevertheless, Text2Shape focused on only two categories from ShapeNet. This limits the generalization of our approach to further shape categories. Besides, due to the lack of current shape-text datasets, we are also unable to validate our approach on further benchmarks. In addition to introducing more datasets, another possible solution is to discover the possibilities of zero-shot text-to-shape generation by distilling the knowledge from 2D vision-language models. Although several works have studied this problem [48], it is still not possible to achieve real sense of flexible text-to-shape generation. In the future, we are going to dig deeper into text-to-shape problems, and search for a possible solution to flexible text-to-shape generation.

**Conclusion.** In this paper, we have focused on the issue of text-to-shape synthesis. We proposed a diffusion model-based framework, *Diffusion-SDF*, to generate voxelized SDFs for shapes conditioned on texts. Generally, we propose a two-stage pipeline for shape generation. Firstly, we propose a patch-wise autoencoder to generate Gaussian SDF representations. Secondly, we introduce the Voxelized Diffusion model together with a novel *UinU-Net* denoiser to generate the patch-independent SDF representations. Further experiments are conducted on several tasks related to text-to-shape synthesis. The results have demonstrated that *Diffusion-SDF* has a strong capability to generate both high-quality and highly diversified 3D shapes.

## A. Qualitative Illustrations

In this section, we provide more qualitative illustrations for the generation results based on our proposed *Diffusion-SDF* approach. Figure 9 shows some extra diversified generated samples for text-conditioned shape generation. Figure 10 displays further text-conditioned shape completion results based on different input shapes. Figure 11 illustrates several text-conditioned shape manipulation results from diverse initial shapes. The extra qualitative illustrations are displayed at the end of this paper.

## References

- [1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas J. Guibas. Learning representations and generative models for 3D point clouds. In *ICML*, pages 40–49, 2018. 1, 2, 5
- [2] Panos Achlioptas, Leonidas J. Guibas, Noah D. Goodman, Judy Fan, and Robert X. D. Hawkins. Shapeglot: Learning language for shape differentiation. In *ICCV*, pages 8937–8946, 2019. 6
- [3] Rohan Chabra, Jan Eric Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard A. Newcombe. Deep local shapes: Learning local SDF priors for detailed 3D reconstruction. In *ECCV*, pages 608–625, 2020. 1, 3
- [4] Paramanand Chandramouli and Kanchana Vaishnavi Gandikota. Ldedit: Towards generalized text guided image manipulation via latent diffusion models. abs/2210.02249, 2022. 5
- [5] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiang Xiao, Li Yi, and Fisher Yu. ShapeNet: An information-rich 3D model repository. abs/1512.03012, 2015. 5, 6
- [6] Kevin Chen, Christopher B. Choy, Manolis Savva, Angel X. Chang, Thomas A. Funkhouser, and Silvio Savarese. Text2Shape: Generating shapes from natural language by learning joint embeddings. In *ACCV*, pages 100–116, 2018. 2, 5, 6, 8
- [7] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J. Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. In *ICLR*, 2021. 2
- [8] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *CVPR*, pages 5939–5948, 2019. 1, 2
- [9] Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3D- $R^2N^2$ : A unified approach for single and multi-view 3D object reconstruction. In *ECCV*, pages 628–644, 2016. 1, 2
- [10] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3D U-Net: Learning dense volumetric segmentation from sparse annotation. In *MICCAI*, pages 424–432, 2016. 4
- [11] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. 2022. 2
- [12] Angela Dai, Charles Ruizhongtai Qi, and Matthias Nießner. Shape completion using 3D-encoder-predictor CNNs and shape synthesis. In *CVPR*, pages 6545–6554, 2017. 1
- [13] Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, pages 8780–8794, 2021. 2
- [14] Yueqi Duan, Haidong Zhu, He Wang, Li Yi, Ram Nevatia, and Leonidas J. Guibas. Curriculum deepsdf. In *ECCV*, pages 51–67, 2020. 1
- [15] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. GET3D: A generative model of high quality 3D textured shapes learned from images. 2022. 1, 5
- [16] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 63(11):139–144, 2020. 2
- [17] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. A papier-mâché approach to learning 3D surface generation. In *CVPR*, pages 216–224, 2018. 1
- [18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4
- [19] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2, 3, 4
- [20] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 5
- [21] Moritz Ibing, Isaak Lim, and Leif Kobbelt. 3D shape generation with grid-based implicit functions. In *CVPR*, pages 13559–13568, 2021. 1
- [22] Chiyu “Max” Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, and Thomas A. Funkhouser. Local implicit grid representations for 3D scenes. In *CVPR*, pages 6000–6009, 2020. 3
- [23] Yue Jiang, Dantong Ji, Zhizhong Han, and Matthias Zwicker. SDFDiff: Differentiable rendering of signed distance fields for 3D shape optimization. In *CVPR*, pages 1248–1258, 2020. 1
- [24] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3D mesh renderer. In *CVPR*, pages 3907–3916, 2018. 1
- [25] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *CVPR*, pages 2416–2425, 2022. 5
- [26] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 2, 3
- [27] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *ICLR*, 2021. 2
- [28] Ke Li and Jitendra Malik. Implicit maximum likelihood estimation. 2018. 2
- [29] Zhengzhe Liu, Yi Wang, Xiaojuan Qi, and Chi-Wing Fu. Towards implicit text-guided 3D shape generation. In *CVPR*, pages 17875–17885, 2022. 2, 5, 6
- [30] Andreas Lugmayr, Martin Danelljan, Andrés Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting

- using denoising diffusion probabilistic models. In *CVPR*, pages 11451–11461, 2022. 5
- [31] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3D point cloud generation. In *CVPR*, pages 2837–2845, 2021. 2
- [32] Lars M. Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *CVPR*, pages 4460–4470, 2019. 1, 2
- [33] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *CVPR*, pages 13482–13492, 2022. 2, 5
- [34] Paritosh Mittal, Yen-Chi Cheng, Maneesh Singh, and Shubham Tulsiani. Autosdf: Shape priors for 3D completion, reconstruction and generation. In *CVPR*, pages 306–315, 2022. 1, 2, 5, 6
- [35] Charlie Nash, Yaroslav Ganin, S. M. Ali Eslami, and Peter W. Battaglia. Polygen: An autoregressive generative model of 3D meshes. In *ICML*, pages 7220–7229, 2020. 1
- [36] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, pages 16784–16804, 2022. 2
- [37] Jeong Joon Park, Peter Florence, Julian Straub, Richard A. Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *CVPR*, pages 165–174, 2019. 1
- [38] Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*, pages 77–85, 2017. 1
- [39] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, pages 5099–5108, 2017. 1
- [40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 5, 6
- [41] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. 2022. 2
- [42] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, pages 8821–8831, 2021. 2
- [43] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *ICML*, pages 1530–1538, 2015. 2
- [44] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, pages 1278–1286, 2014. 3
- [45] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10674–10685, 2022. 2, 5, 7
- [46] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015. 2
- [47] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. 2022. 2
- [48] Aditya Sanghi, Hang Chu, Joseph G. Lambourne, Ye Wang, Chin-Yi Cheng, Marco Fumero, and Kamal Rahimi Malekshah. Clip-forge: Towards zero-shot text-to-shape generation. In *CVPR*, pages 18582–18592, 2022. 2, 5, 6, 8
- [49] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pages 2256–2265, 2015. 2, 3
- [50] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 5
- [51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 4
- [52] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In *NeurIPS*, pages 82–90, 2016. 2
- [53] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3D ShapeNets: A deep representation for volumetric shapes. In *CVPR*, pages 1912–1920, 2015. 1
- [54] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomír Mech, and Ulrich Neumann. DISN: deep implicit surface network for high-quality single-view 3D reconstruction. In *NeurIPS*, pages 490–500, 2019. 1, 2
- [55] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3D object reconstruction without 3D supervision. In *NeurIPS*, pages 1696–1704, 2016. 2
- [56] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge J. Belongie, and Bharath Hariharan. Pointflow: 3D point cloud generation with continuous normalizing flows. In *ICCV*, pages 4540–4549, 2019. 1, 2
- [57] Xumin Yu, Yongming Rao, Ziyi Wang, Zuyan Liu, Jiwen Lu, and Jie Zhou. PointR: Diverse point cloud completion with geometry-aware transformers. In *ICCV*, pages 12478–12487, 2021. 5
- [58] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. LION: latent point diffusion models for 3D shape generation. 2022. 2
- [59] Linqi Zhou, Yilun Du, and Jiajun Wu. 3D shape generation and completion through point-voxel diffusion. In *ICCV*, pages 5806–5815, 2021. 2, 5

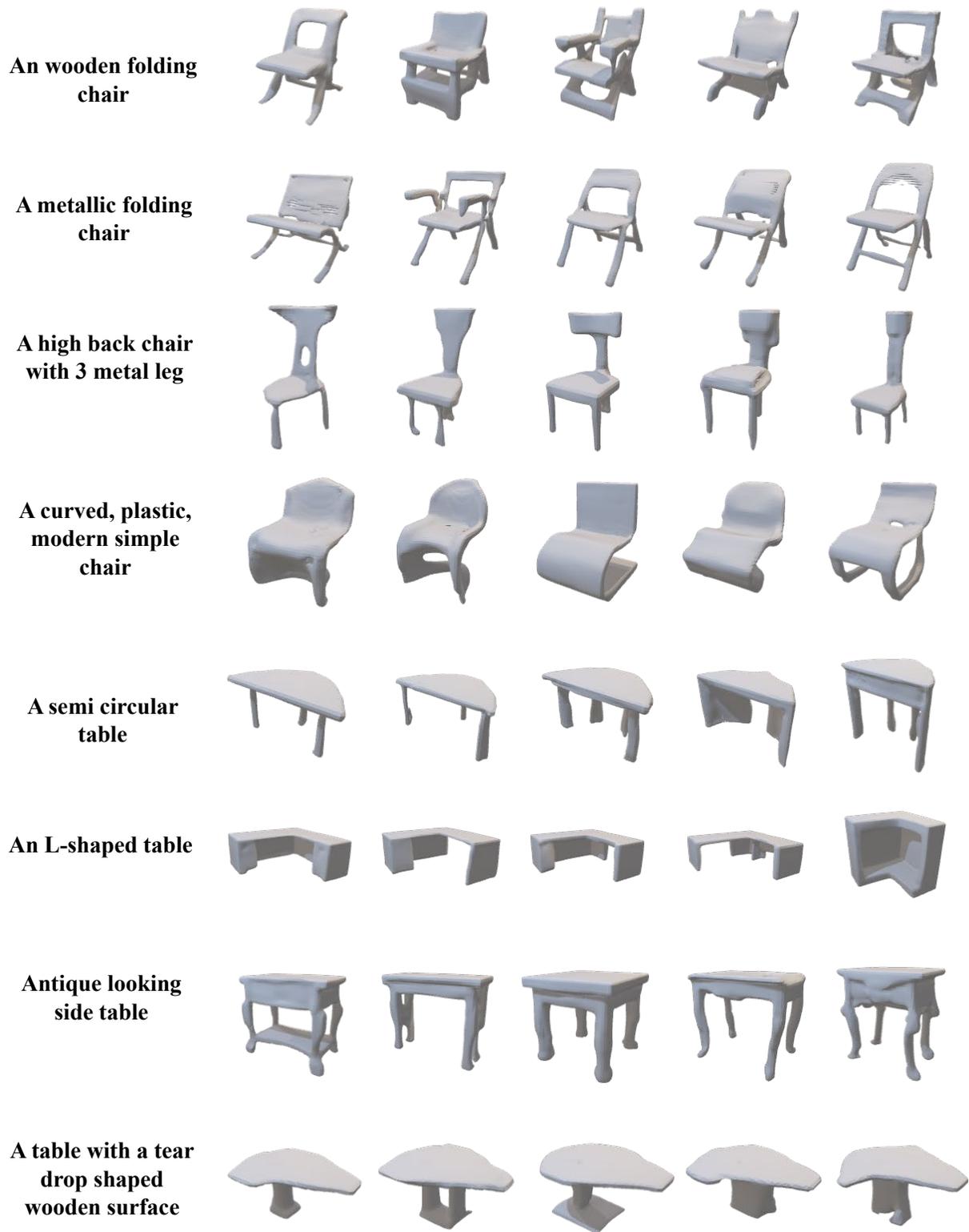


Figure 9. More illustrations on diversified text-to-shape generation.

**Given shape**

**A sofa**

**A ball chair**

**A straight chair**

**A toilet**



Figure 10. More illustrations on text-conditioned shape completion.

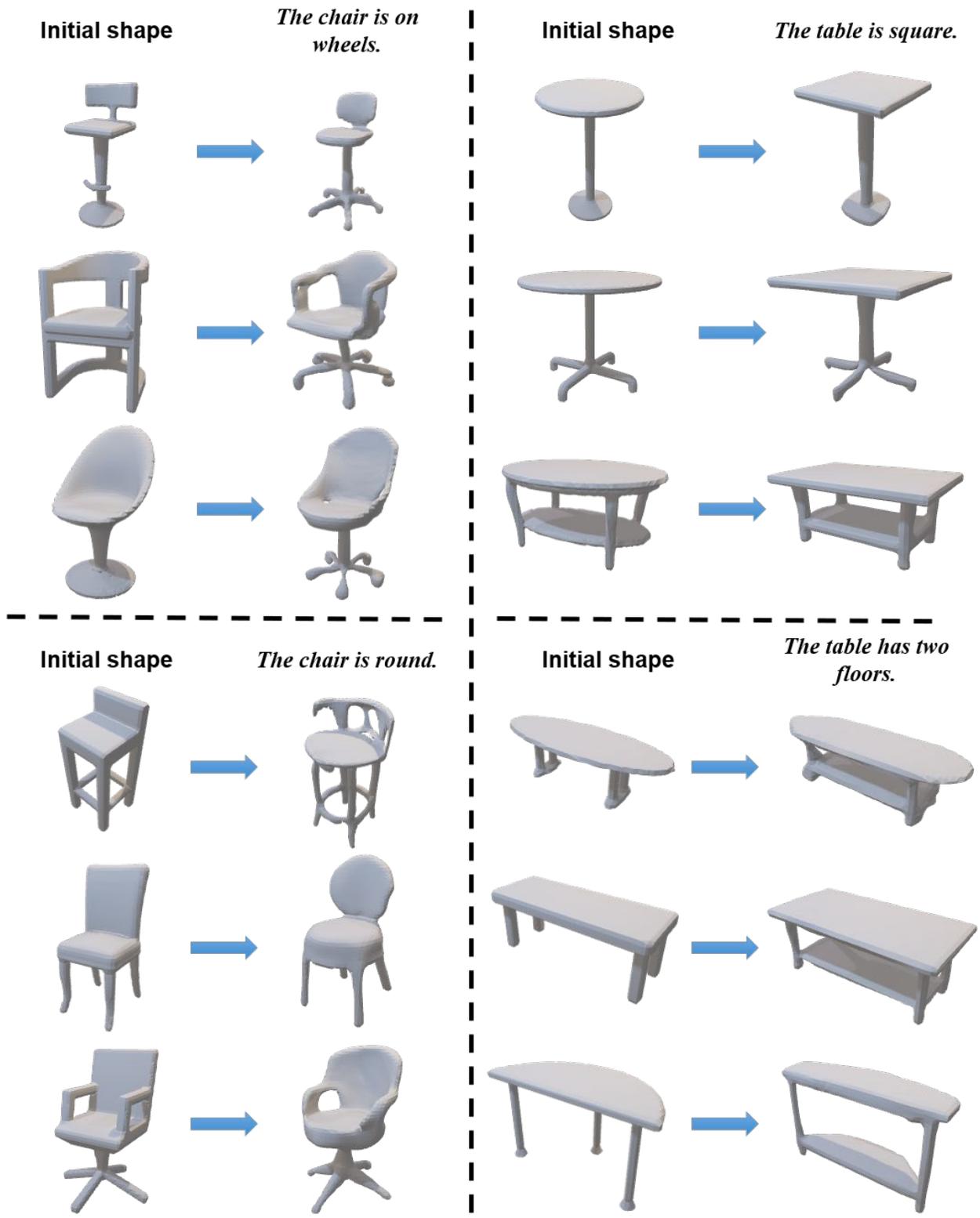


Figure 11. More illustrations on text-conditioned shape manipulation.