This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Order Matters: 3D Shape Generation from Sequential VR Sketches

Yizi Chen [1*] Sidi Wu [1*] Tianyi Xiao [1] Nina Wiedemann [1] Loic Landrieu[2]

[1] ETH Zurich [2] LIGM, ENPC, IP Paris, Univ Gustave Eiffel, CNRS

Abstract

VR sketching lets users explore and iterate on ideas directly in 3D, offering a faster and more intuitive alternative to conventional CAD tools. However, existing sketch-to-shape models ignore the temporal ordering of strokes, discarding crucial cues about structure and design intent. We introduce VRSketch2Shape*, the first framework and multi-category dataset for generating 3D shapes from* sequential *VR sketches. Our contributions are threefold: (i) an automated pipeline that generates sequential VR sketches from arbitrary shapes, (ii) a dataset of over* 20k *synthetic and* 900 *hand-drawn sketch–shape pairs across four categories, and (iii) an order-aware sketch encoder coupled with a diffusion-based 3D generator. Our approach yields higher geometric fidelity than prior work, generalizes effectively from synthetic to real sketches with minimal supervision, and performs well even on partial sketches. All data and models are released open-source on* https://chenyizi086.github.io/VRSketch2Shape_website/.

1. Introduction

Creating high-quality 3D content is central to architecture and industrial design and is well supported by powerful CAD tools such as Blender [35, 68]. However, these tools have a steep learning curve and are optimized for precision, making them ill-suited for rapid ideation and early-stage exploration—key steps in the creative process. Recent research has therefore explored text-conditioned generative models for 3D shape synthesis [16, 29, 54], but natural language remains too ambiguous to specify complex geometries [55, 76].

Sketch-Based 3D Design. Sketching provides a fast and intuitive way to express spatial concepts. Early work relied on single- or multi-view *2D* sketches for shape generation [3, 18, 78, 81]. With the advent of commodity VR/AR systems, *3D* sketching has emerged as a natural and immersive alternative [8, 41]: drawing directly in 3D space eliminates perspective ambiguities and occlusions inherent to 2D sketches, while enjoying a more natural and immersive design experience.

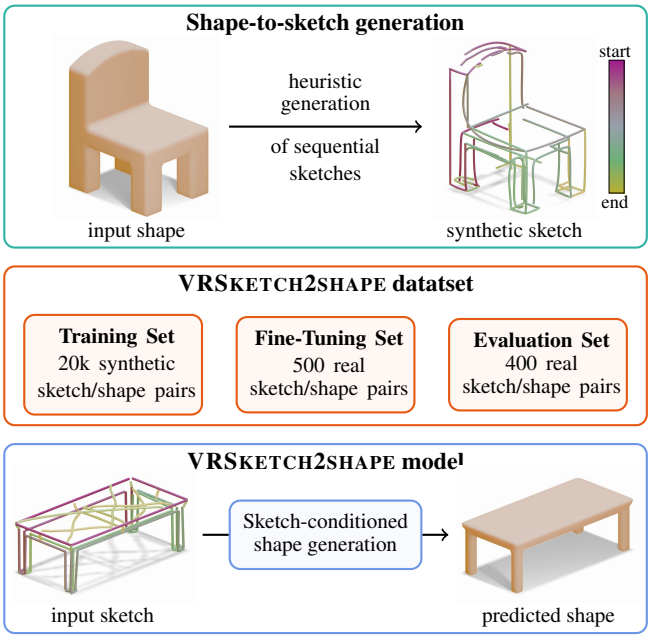


Figure 1. **Overview of Contributions.** We propose: (i) a learning-free pipeline to generate realistic sequential 3D sketches from arbitrary shapes; (ii) the open-access VRSketch2Shape dataset, with 20k synthetic and 900 hand-drawn sketch–shape pairs; and (iii) an order-aware, diffusion-based model to generate high-fidelity 3D shapes from sequential VR sketches.

Open Challenges. Despite its promise, VR sketch–conditioned shape generation faces three main challenges: (i) **Data scarcity.** Collecting paired VR sketches and 3D meshes is costly; the only public benchmark [40] includes just 1,005 sketch–chair pairs from a single category. (ii) **Geometric misalignment.** Human-annotated sketches naturally include spatial inaccuracies from perspective and depth-perception errors, resulting in imperfect alignment with target shapes and complicating both training and evaluation. (iii) **Temporal information loss.** Existing pipelines to create shapes from VR sketches [8, 40] but treat

*Equal contribution.

them as unordered point clouds, thus discarding stroke order and length; yet these signals encode important information about connectivity, structure, and design intent.

VRSKETCH2SHAPE. We propose a new framework to generate 3D shapes from *sequential VR sketches*. We model a sketch as a sequence of *strokes*, each itself an ordered sequence of 3D *points*. Building on this formulation, we make three primary contributions:

- **Synthetic Sketch Generation.** An automatic pipeline that produces sequential sketches from arbitrary 3D shapes, yielding over 20k paired samples for large-scale training.
- **Real Sketch Collection.** A custom VR sketching interface with surface snapping to reduce drawing errors. Using this tool, we produced 900 VR sketches across four categories, each annotated with complete stroke and point ordering.
- **Order-Aware Shape Generation.** A sketch encoder that models stroke sequences using a modified BERT architecture [15], coupled with SDFUSION [11] for diffusion-based shape generation.

Results. The VRSKTECH2SHAPE model outperforms prior work by a large margin on both existing and newly collected benchmarks. Trained solely on synthetic sketches, it generalizes effectively to real sketches with little or no fine-tuning, highlighting both the robustness of our model and the utility of our synthetic data. Moreover, the model remains stable with partial sketches, enabling cross-modal shape completion—an ability that could greatly accelerate interactive 3D design workflows.

2. Related work

We first review prior work on 3D shape generation from conventional modalities such as text and images (Sec. 2.1), followed by sketch-based methods (Sec. 2.2). We then discuss related approaches for sketch generation and encoding (Sec. 2.3).

2.1. Classical 3D Shape Generation

Generative Approaches. Early work on 3D shape generation explored a range of paradigms, including Generative Adversarial Networks (GANs) [1, 10, 70, 71, 80], Variational Autoencoders (VAEs) [12, 48], and auto-regressive models [46, 74]. Recent advances have shifted toward diffusion-based approaches, which produce high-fidelity 3D content in the form of point clouds [25, 42], voxel occupancy grids [82], or meshes [36].

Implicit Representations. Unlike explicit 3D formats, implicit neural fields offer continuous surfaces, compact storage, and theoretically infinite resolution. Recent efforts have used diffusion methods to generate signed distance functions (SDFs) [11, 47] or neural radiance fields [44, 49]. To improve scalability, several diffusion models operate in latent space [11, 47, 52]. Generation can be conditioned by images [11, 33, 34, 56, 59, 61] or text prompts [9, 11, 16, 29], enabling controllable 3D shape synthesis.

Table 1. **3D Sketch Datasets.** VRSKETCH2SHAPE is the first open-access collection that spans multiple object categories and includes both synthetic and real VR sketches. CD measures the asymmetric Chamfer distance between the real sketch and shapes.

	open-access	categories	number of sketches	CD $\times 1000$ Sk $\mapsto$ Sh
3DVRChair [40]	✓	1	1005 real	55.6
KO3D+ [8]	✗	6	4,200 real	-
VRSS [17]	✗	55	2097 real	-
VRSKETCH2SHAPE (ours)	✓	4	20,838 synthetic + 900 real	5.5

2.2. Sketch-Based 3D Shapes Generation

2D sketches. Sketching has recently emerged as a powerful modality for 3D shape generation and editing, enabling users to specify geometry through intuitive freehand input. Early work learned deterministic mappings from sketches to 3D shapes [6, 22, 23, 67, 77], whereas recent methods adopt diffusion-based generative models [3, 81]. To mitigate the inherent ambiguity of single-view sketches, some approaches incorporated camera parameters or viewpoint conditioning [7, 18, 78, 81], while others leveraged multi-view sketches for higher geometric accuracy [14, 39].

VR sketches. More recently, *3D* sketches drawn in virtual reality (VR) have been explored for shape generation, providing a more immersive and spatially intuitive design interface [8, 17, 40, 41]. Existing methods represent VR sketches as 3D point clouds and align their latent representations with those of point clouds sampled from target shapes [41] or with 2D renderings of 3D shapes [8]. However, these approaches ignore the sequential nature of sketches, whereas the VRSKETCH2SHAPE explicitly models the temporal order of strokes and points.

2.3. VR Sketch Synthesis and Representation

Classical 3D Sketch Synthesis. Rendering sketches of 3D shapes is a long-standing problem in computer graphics and vision. Classical methods primarily aim at perceptual realism [26] or stylistic expressiveness [21, 69]. In contrast, our goal is not to produce visually realistic sketches, but to extract structurally faithful VR lines that can serve as supervision for the inverse sketch-to-shape problem. Consequently, design choices in our pipeline are guided by the

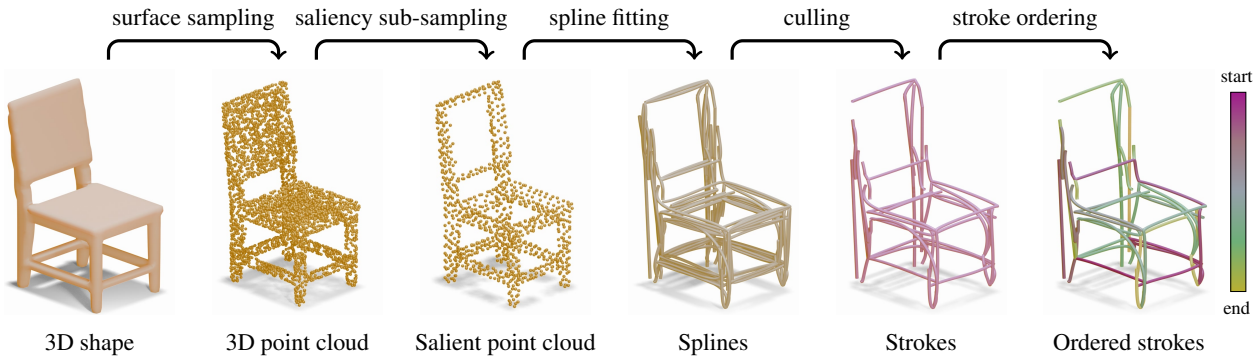


Figure 2. **Synthetic Sketch Generation.** We propose a heuristic, learning-free pipeline for generating 3D sequential sketches from 3D shapes. We first uniformly sample points on the surface and retain only *salient* points. Bézier splines are then fitted through these points to form candidate strokes, which are subsequently merged and simplified. Finally, we order both points and stroke to obtain temporally sequential 3D sketches.

performance of downstream models trained on the resulting synthetic sketches rather than by their visual quality.

Related lines of work study procedural CAD [24] or boundary-representation models [73], where shapes are defined through parametric modeling operations (*e.g.*, extrusion, revolution) applied to geometric primitives and curves (*e.g.*, cylinders, splines). While such representations also encode sequential geometric structure, they assume structured CAD inputs, whereas our setting targets informal VR sketches produced during early-stage ideation.

Deep Sketch Synthesis. Early work on sketch generation focused on 2D sketches, using image-to-image translation networks [28, 58], VAEs [19], or auto-regressive models [4]. These approaches typically required large datasets of human sketches for supervision. To mitigate this limitation, subsequent methods directly optimized vectorized representations under the guidance of pretrained vision–language [51, 64, 65] or diffusion models [2, 72]. Other works addressed sketch completion using GANs [32] or transformers [30]. Building on these advances, recent approaches extend sketch generation to 3D, synthesizing parametric curves from text, single-view, or multi-view images and optimizing their parameters via pretrained image models [13, 66, 79]. In contrast, our synthetic sketch generation pipeline relies purely on geometric heuristics and is entirely training-free.

Sketch Encoding. 2D sketches are typically processed either as images using convolutional networks [13, 18, 75, 81], or as sequences using recurrent or transformer-based models [63] to capture their temporal and structural logic [20, 30]. In contrast, most existing approaches for 3D VR sketches still represent them as unordered point clouds [8, 17, 40] and apply point-based encoders such as PointNet++ [50], thereby discarding the intrinsic stroke order. However, VR sketches are inherently sequential, as the drawing order encodes meaningful cues about connectivity, structure, and design intent. In this work, we encode VR sketches directly as ordered sequences of 3D points, allowing our model to exploit both their spatial geometry and the temporal dependencies of the sketching process.

3. VRSketch2Shape Dataset

We introduce VRSketch2Shape, a dataset of real and synthetic sequential VR sketches. We first describe the collection of 900 real VR sketches aligned with ShapeNet models (Sec. 3.1), and then detail our automatic synthetic sketch generation pipeline (Sec. 3.2).

Setup. We define a VR sketch as a collection of *3D polylines* (or *strokes*), each represented by a temporally ordered sequence of 3D points drawn in a single continuous motion. Our dataset preserves both stroke order and point order, providing temporal information often discarded in prior work that treats sketches as unordered point sets.

3.1. Real Data Collection

We built a Unity-based VR interface that allows participants to visualize and sketch directly over a reference 3D model. A key challenge in VR sketching is depth ambiguity: without guidance, users often draw strokes that float in front of or behind the surface, resulting in imprecise and hard-to-use annotations. To mitigate this issue, we implemented a *surface-snapping* mechanism that projects each drawn point onto the underlying 3D model along the shortest path, ensuring geometric alignment between the sketch and the object. As shown in Tab. 1, this snapping step produces sketches that are substantially more faithful to the input shapes—as measured by asymmetric sketch-to-shape Chamfer distance, leading to a more reliable and less noisy benchmark for sketch-conditioned generation.

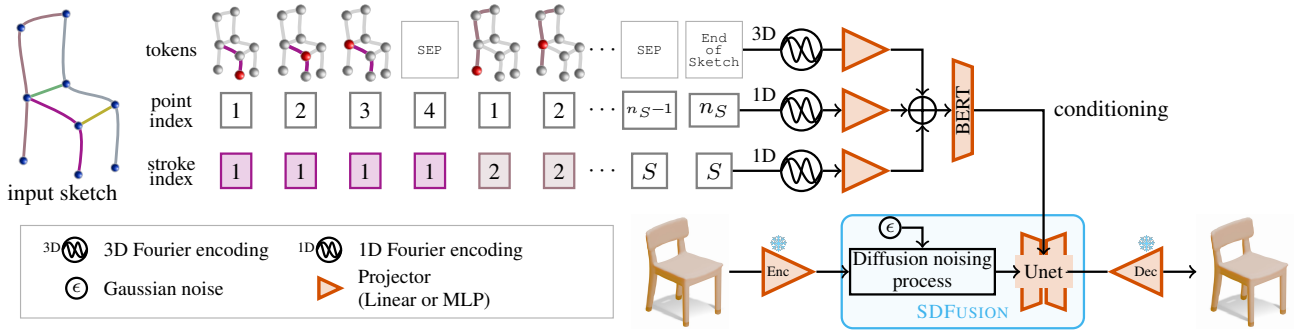


Figure 3. **VRSketch2Shape Model.** An input VR sketch is tokenized into a sequence of points organized along ordered strokes. Each 3D point is encoded using 3D Fourier features and an MLP, while stroke and point indices are encoded with 1D Fourier features followed by a linear projection. The resulting embeddings are summed and passed through a lightweight BERT encoder. The encoded token sequence is then used to condition SDFusion, a diffusion-based 3D shape generation model.

We recruited 15 participants, who completed a short tutorial before sketching multiple objects from ShapeNet. The resulting dataset contains 900 sketches across four categories: 300 chairs, 200 tables, 200 cabinets, and 200 airplanes. With an average of 15 minutes per sketch, data collection required approximately 225 person-hours.

3.2. Synthetic Sketch Generation.

Because collecting real sketches is expensive and tedious, we propose a fully automatic pipeline to generate synthetic VR sketches from 3D meshes (Fig. 2), producing 20,838 samples in roughly 10 hours on a standard work station.

Extracting Salient Points. We begin by uniformly sampling 2048 points on the surface of the input mesh. Sketches typically emphasize visually prominent geometric features such as edges, corners, and holes. We emphasize regions of high curvature and structural significance by extracting the *salient point cloud* using Sharp Edge Sampling (SES) [5] and a curvature threshold of 15.

Recovering Strokes. We then fit Bézier splines to the salient point cloud using EMAP [27] with a maximum degree of 2 and minimum segment length of 12. The points along each spline form the individual strokes. Next, we apply a culling stage inspired by the approach proposed by Liu *et al*. [31]. We first remove redundant points in near-linear segments with a cosine distance threshold of 0.04. Finally, we merge strokes whose endpoints lie within a threshold of 2% of the normalized shape size.

Ordering Strokes. To approximate human drawing order, we connect stroke endpoints based on spatial proximity and perform a depth-first traversal of the resulting connectivity graph. We introduce stochasticity by skipping nearest connections with a probability of 10%, yielding coherent yet varied stroke sequences.

Dataset Structure. The proposed VRSketch2Shape dataset is organized into four parts:

- **Synthetic training set:** 20,838 sketch–shape pairs generated with our automatic pipeline.
- **Real fine-tuning set:** 500 sketch–shape pairs (200 chairs and 100 per other category) for domain adaptation from synthetic to real sketches.
- **Real evaluation set:** 400 pairs (100 per category) reserved for final quantitative and qualitative evaluation.

4. VRSketch2Shape Model

In this section, we present our model for generating 3D shapes from sequential VR sketches. The overview is shown in Fig. 3. We first describe our sketch encoder based on BERT (Sec. 4.1), then explain how it interfaces with the diffusion-based shape generator SDFusion (Sec. 4.2).

4.1. Encoding 3D Sketches

We treat each sketch as a sequence and encode it with a transformer-based architecture inspired by BERT [15]: the sketch is tokenized, embedded, enriched with positional encodings, and processed by several transformer blocks.

Sketch tokenization. A VR sketch consists of an ordered set of strokes, each formed by a sequence of 3D points. We introduce two special tokens: `SEP`marks the end of a stroke, and `EoS` (End of Sketch) marks the end of the entire sketch. A sketch $\mathcal{S}$ with S strokes is thus tokenized as:

$$\mathcal{S} = \left[p_1^1, \cdots, p_{n_1}^1, \texttt{SEP}, \cdots, p_1^S, \cdots, p_{n_S}^S, \texttt{SEP}, \texttt{EoS} \right], \quad (1)$$

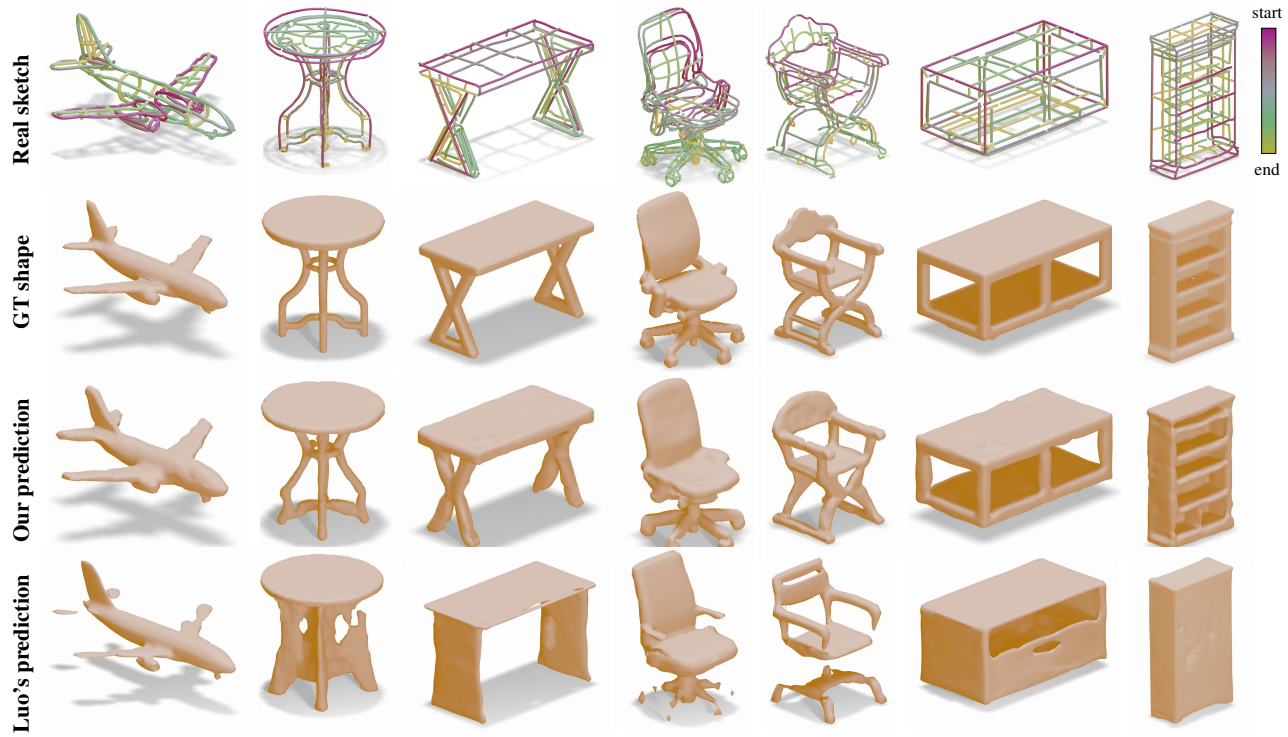


Figure 4. **Qualitative Illustrations.** Comparison between our method and Luo *et al*. [41] on the real test set of VRSKETCH2SHAPE. Both models are pretrained on the same synthetic sketches and fine-tuned on real data. Our approach generates shapes that are more detailed, structurally accurate, and topologically faithful to the target geometry.

where n_s is the number of points in stroke s, and each point $p = (x, y, z) \in [0, 1]^3$ stores normalized 3D coordinates. We denote by p_i^s the i-th point of the s-th stroke.

Spatial embedding. Following Mildenhall *et al*. [45], we map each coordinate of $p = (x, y, z)$ through a Fourier feature encoding, known to better capture high-frequency geometric details:

$$\Phi_{\text{spa}}(t) = \left[\sin(2^\ell \pi t), \cos(2^\ell \pi t)\right]_{\ell=0}^{L-1} \in \mathbb{R}^{2L}, \quad (2)$$

where L is the number of frequency bands and $[\cdot, \cdot]$ denotes the feature-wise concatenation operator. We concatenate the encoded coordinates and map them to the model dimension D with $\text{MLP}_{\text{spa}} : \mathbb{R}^{6L} \mapsto \mathbb{R}^D$:

$$E_{\text{spa}}(p) = \text{MLP}_{\text{spa}}\left(\left[\Phi_{\text{spa}}(x), \Phi_{\text{spa}}(y), \Phi_{\text{spa}}(z)\right]\right) . \quad (3)$$

The embeddings of the separator tokens $E_{\text{spa}}(\texttt{SEP})$ and $E_{\text{spa}}(\texttt{EOS})$ are learned as free parameters in $\mathbb{R}^D$.

Sequence Embeddings. Order matters at two levels: stroke index s and within-stroke point index i. We encode both positions using the sinusoidal encoding from the original Transformers [63]:

$$\Phi_{\text{seq}}(t) = \left[\sin\left(\frac{t}{10{,}000^{2d/D}}\right), \cos\left(\frac{t}{10{,}000^{2d/D}}\right)\right]_{d=0}^{D/2-1} . \quad (4)$$

Stroke and point embeddings are obtained with the linear projections $\text{Lin}_{\text{stroke}}$ and $\text{Lin}_{\text{point}} : \mathbb{R}^D \mapsto \mathbb{R}^D$:

$$E_{\text{stroke}}(s) = \text{Lin}_{\text{stroke}}\left(\Phi_{\text{seq}}(s)\right) \quad (5)$$

$$E_{\text{point}}(i) = \text{Lin}_{\text{point}}\left(\Phi_{\text{seq}}(i)\right) . \quad (6)$$

Final Token Embedding. For a point token p_i^s, we sum the spatial, stroke, and point embeddings:

$$E(p_i^s) = E_{\text{spa}}(p_i^s) + E_{\text{stroke}}(s) + E_{\text{point}}(i) . \quad (7)$$

Augmentation strategies. We apply the following sketch-specific stochastic data augmentations during training:

- **Stroke dropping.** Randomly mask 15% of the strokes.
- **Point dropping.** Randomly mask 30% of the points within the remaining strokes.
- **Stroke swapping.** Randomly swap 20% of the strokes with another stroke in the sketch.

The masked tokens are replaced with a learnable token $\texttt{MASK}$ such that $E_{\text{spa}}(\texttt{MASK}) \in \mathbb{R}^D$

Differences From SketchBERT. While our encoder shares structural similarities with the two-dimensional SKETCHBERT [30], it differs in three key aspects: (i)

Table 2. **Quantitative results.** Comparison of sketch-to-shape generation methods on the public 3DVRCHAIR dataset and our proposed VRSKETCH2SHAPE dataset. ⋆ use 2D renders of sketches.

	3DVRChair [40]		VRSketch2Shape (ours)			
			chair only		all categories	
	F-score ↑	$CD_{\times 1000}$↓	F-score ↑	$CD_{\times 1000}$↓	F-score ↑	$CD_{\times 1000}$↓
LAS-diffusion⋆ [81]	26.1	66.0	37.0	51.1	40.2	27.1
Luo *et al*. [41]	26.6	35.5	42.2	13.4	48.8	13.0
VRSketch2Shape (ours)	**31.1**	**25.8**	**64.3**	**4.0**	**69.8**	**4.8**

Point coordinates are represented via spatial Fourier features rather than raw positions, (ii) Stroke delimiters are treated as learned tokens instead of concatenated one-hot flags, and (iii) Continuous Fourier-based encodings replace fixed lookup tables, allowing flexible handling of variable-length and user-dependent sketch styles.

4.2. Diffusion-Based Shape Generation

We condition 3D shape generation on the sketch embeddings using SDFUSION [11], a latent diffusion model originally designed for text- and image-guided shape synthesis. Our sketch encoder interfaces directly with the diffusion model: the sequence of tokens produced by the BERT-style encoder serves as the conditioning input to SDFUSION.

During training, each ground-truth 3D shape is first voxelized and encoded into a compact latent representation using a pretrained 3D VQ-VAE [62]. Gaussian noise is then added to this latent through the forward diffusion process, and a U-Net [53] is trained to predict the denoised latent. The VQ-VAE remains frozen, while the U-Net and our sketch encoder are optimized jointly using an ℓ_2 reconstruction loss between the predicted and target latents, following the approach of [52]. At inference time, we encode the sketch and apply the reverse denoising process with 100 DDIM steps [57] starting from random noise to synthesize the corresponding 3D shape. Unlike previous approaches that require multi-stage training or modality alignment steps, our framework is trained end-to-end in a single stage.

Implementation Details. We encode spatial coordinates using Fourier features with $L = 10$ frequencies per axis, concatenated and projected to a $D = 256$-dimensional embedding through a 2-layer MLP with 256 hidden units. The BERT-style transformer has 6 layers, 8 attention heads, and a feed-forward with an inner width ratio of 1.

We train all models with AdamW [38] (default parameters) using a base learning rate of 10^{-4} and ReduceLROnPlateau decay with a patience of 10 epochs and decay of 0.5. We use a batch size of 16 for synthetic pretraining and 12 for real-data fine-tuning. The model is pretrained for 200 epochs on synthetic sketches and optionally fine-tuned for 300 epochs on real sketches. A dropout rate of 0.1 is applied in the transformer encoder.

5. Numerical Experiments

We present our evaluation setting (Sec. 5.1), our results (Sec. 5.2), and finally an ablation study (Sec. 5.3).

5.1. Datasets and Evaluation Metrics

We evaluate our approach on two datasets: 3DVRCHAIR [40] and our proposed VRSKETCH2SHAPE.

3DVRCHAIR [40]. This dataset is the only other publicly available benchmark for VR sketch–based 3D generation. It contains 1,005 real sketch–shape pairs from the *chair* category only. We use the official split of 803 samples for training and 202 for evaluation.

VRSKETCH2SHAPE dataset. We also evaluate on our proposed datasetby first training on the synthetic subset and evaluating under two adaptation protocols:

- **Zero-shot adaptation.** The model is evaluated directly on the real sketch test set to assess the synthetic-to-real generalization gap.
- **Few-shot adaptation.** The model is fine-tuned on a subset or the entirety of the fine-tuning set.

Metrics. Following standard practice [40], we evaluate the generated 3D shapes using two metrics:

- **Chamfer Distance (CD).** We uniformly sample $N = 4096$ points from the ground-truth surface and N points from the generated surface, and compute the mean bidirectional distance.
- **F-score.** We evaluate the geometric accuracy of generated shapes using the F-score, which combines precision and recall [60]. We use a threshold $\delta = 0.02$ to reflect the inherent imprecison of manual sketching.

For all metrics, we follow the evaluation protocol of [40]: the predicted shapes are first aligned to the ground truth by translating and scaling them such that the diagonal of their bounding box coincides.

5.2. Results and Analysis

We report quantitative results in Tab. 2, comparing our approach with all publicly available baselines. We conduct four main experiments.

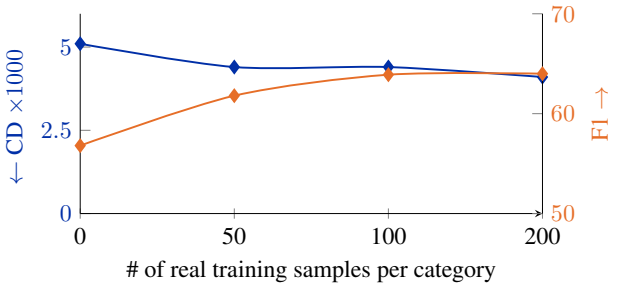


Figure 5. **Few-Shot Adaptation.** Chamfer Distance and F1-score on our real test set as a function of the number of real sketches used to fine-tune a model pretrained on synthetic data.

Experiment on 3DVRCHAIR. We train our model on the training split of 3DVRCHAIR and evaluate on its test set. We compare against the official pretrained weights of Luo *et al*. [40] as well as a retrained 2D-based LAS-Diffusion baseline (see Sec. 5.3 for implementation details). As shown in Tab. 2, our method outperforms competing approaches by a large margin, reducing CD by more than 60% and improving the F-score by over 40% on our dataset. We were unable to evaluate Chen *et al*. [6] due to the absence of publicly released code or checkpoints, and their reported results rely on an unspecified evaluation protocol, making direct comparison impossible.

Experiment on VRSKETCH2SHAPE. We train all models on our synthetic subset, fine-tune on the real fine-tuning subset, and evaluate on the held-out real test set. We report results both on the *chair* category (for comparability with 3DVRCHAIR) and across all four categories. Our method achieves the best performance in both settings, confirming the benefit of our proposed model. Interestingly, Luo *et al*. also perform better on our dataset than on theirs, likely due to the reduced ambiguity of our automatically aligned synthetic sketches. These findings jointly validate the effectiveness of our model and the utility of our dataset for robust VR sketch–based shape generation.

We visualize in Fig. 4 representative synthetic and real sketches from our test set, along with 3D reconstructions predicted by our model and by Luo *et al*. The generated shapes closely match the ground-truth geometry and preserve object topology and details more faithfully. We note that our reconstructions sometimes appear slightly oversmooth, which we attribute to optimizing only in the latent space of a pretrained, frozen 3D VQ-VAE.

Few-shot Synthetic-to-Real Adaptation. We evaluate in Fig. 5 the effect of fine-tuning our model trained on synthetic sketches with real sketches. We take our model pretrained on synthetic data only, and fine-tune it (or not). Performance improves steadily as more real sketches are introduced, but as few as 50 sketches per category suffice to reach near-optimal results. Remarkably, even in the zero-shot setting (no fine-tuning), our model performs strongly, demonstrating that the synthetic sketches generated by our heuristic pipeline provide effective supervision for real-world generalization.

Table 3. **User Study.** Likert scale: *5-Excellent*: faithful geometry; *4-Good*: minor artifacts; *3-Acceptable:* recognizable, missing details; *2-Poor*: weak correspondence; *1-Failed*.

	w. reference	free-hand
Luo's prediction	2.76 ± 0.84	2.02 ± 0.89
Ours prediction	$\mathbf{3.92 \pm 0.74}$	$\mathbf{3.60 \pm 1.01}$
Reference shape	4.76 ± 0.50	-

Free-hand Sketches. Real creative workflows involve *free-hand* sketching, *i.e*. users draw without any reference model. This raises an important question: *Can a model trained primarily on synthetic and reference-guided sketches generalize to free-hand inputs?* We conducted a user study evaluating sketches drawn with and without reference shapes, comprising 40 sketches: 20 free-hand and 20 reference-guided. From these, we constructed 100 sketch-shape pairs: 40 reconstructions generated by our model, 40 by Luo *et al*., and the 20 ground-truth shapes used for the reference-guided sketches. A total of 38 participants were each shown 50 randomly ordered sketch-shape pairs, yielding 1,900 ratings, in which sketch-shape correspondence was evaluated on a 1-5 Likert scale. As reported in Tab. 3, our method significantly outperforms Luo *et al*. and maintains strong performance on free-hand sketches. Fig. 6 shows that our model's robust performance on free-hand sketches.

5.3. Ablation Study

We perform an ablation study to quantify the contribution of our main design choices, summarized in Tab. 4. All experiments are conducted in a zero-shot synthetic-to-real setting on the *chair* subset of our dataset.

Impact of Design Choices. We first assess the importance of key components of our model:

- **w/o ordering.** We remove stroke and point indices from the sketch encoder, keeping only E_{spa} while discarding E_{stroke} and E_{point}. Performance drops sharply, confirming that *order does matter*.
- **w/o augmentations.** Disabling augmentations noticeably degrades performance, showing that these simple augmentations effectively improve robustness and prevent overfitting.
- **w/o pre-training.** We skip pre-training on synthetic data and train only on the 200 real sketch–shape pairs from the fine-tuning chair set. The model collapses to trivial solutions, emphasizing the necessity of large-

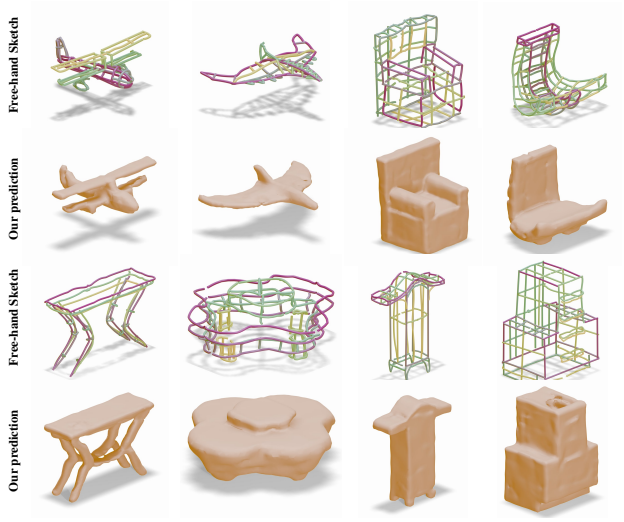


Figure 6. **Shape Generation from Free-Hand Sketches.** Our model generalizes well to free-hand sketches drawn without any reference shape for airplanes, chairs/sofas, tables, and cabinets, producing detailed and plausible reconstructions that reflect the user's intent.

scale synthetic pre-training and the effectiveness of our fully automated sketch synthesis pipeline. Reaching a comparable data volume through manual sketching would require prohibitive human effort.

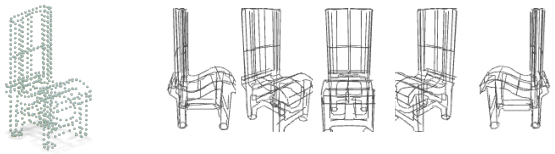
- **SketchBERT encoder.** Replacing our encoder with a direct 3D extension of SketchBERT [30] leads to a substantial drop in accuracy, highlighting the importance of our design adaptation for 3D sequential data.

Impact of Sketch Format. We then compare representing sketches as ordered point–stroke sequences versus other common encodings:

- **Sketches as point clouds.** We uniformly sample 1,024 points along all strokes and encode them using PointNet++ [50], following Luo *et al.* [41]. The clear performance degradation indicates that our sequential formulation, and not just the diffusion generator, accounts for much of the observed improvement.
- **Sketches as images.** We convert each 3D sketch into a mesh and render five 2D views using `pyrender` [43]. Following LAS-Diffusion [81], the rendered images are encoded with a pretrained VGG network. The embeddings and corresponding camera poses condition a latent diffusion model trained to predict 64^3 occupancy grids. For fairness, we retrain this model on our full synthetic dataset and apply the pretrained super-resolution module from [81] to upsample predictions to 128^3 signed distance fields. This variant yields a marked performance drop, as occlusions in the rendered views cause missing or distorted geometry.

Table 4. **Ablation study.** We evaluate variants of our models trained on synthetic sketches and evaluated on real *chair* sketches to show the contribution of each component. Discarding stroke order, augmentations, or pretraining significantly degrades performance, while alternative sketch format (point clouds or multiview) fail to capture 3D sequential structure effectively.

Method	F-score ↑	CD×1000 ↓
Full model	**56.8**	**5.1**
w/o stroke ordering	48.9	7.1
w/o augmentation	49.0	6.6
w/o pretraining	12.0	99.9
SketchBert encoding	50.4	6.3
Sketches as point clouds (a)	30.8	25.8
Sketches as Images (b)	23.8	62.6



(a) Sketches as point clouds (b) Sketches as images

Generation Speed. On a single consumer NVIDIA RTX 4090 GPU, generating a 3D shape from a sketch takes on average 6.61 ±1.18 s, including 26 ms for sketch encoding, 6.55 s for latent denoising, and 31 ms for SDF decoding with Marching Cubes [37]. Training on our 20k+ synthetic sketches completes in roughly 50 hours, and fine-tuning on the real sketches adds an additional 10 hours.

Limitations. Since our model is only supervised by its ability to denoise latent embeddings and uses a frozen 3D VQ-VAE, reconstruction quality and inference speed is ultimately bounded by the capacity of this encoder–decoder. In particular, the 64^3 SDF resolution limits fine-grained geometric detail. Future work could lift this constraint by training the shape generator end-to-end at higher spatial resolutions.

6. Conclusion

We introduced VRSKETCH2SHAPE, the first open-source, multi-category dataset and model for 3D shape generation conditioned on sequential VR sketches. Our contributions include an automatic pipeline for scalable synthetic sketch generation, a curated collection of real hand-drawn sketches with preserved drawing order, and a stroke-aware sketch encoder coupled with a diffusion-based shape generator. Extensive experiments show that explicitly modeling stroke order improves structural fidelity and generalization, enabling effective training even on synthetic data alone.

Acknowledgment

We thank Gege Gao for insightful discussions and Prof. Lorenz Hurni for their support and access to GPUs. This work is supported by Hi! PARIS and ANR/France 2030 program (ANR-23-IACL-0005). The data collection process was funded by the Swiss National Science Foundation (SNSF) under the grant *3D Sketch Maps* (Grant No. 202284).

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3D point clouds. In *ICML*, 2018. 2

[2] Ellie Arar, Yarden Frenkel, Daniel Cohen-Or, Ariel Shamir, and Yael Vinker. SwiftSketch: A diffusion model for image-to-vector sketch generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, 2025. 3

[3] Hmrishav Bandyopadhyay, Subhadeep Koley, Ayan Das, Ayan Kumar Bhunia, Aneeshan Sain, Pinaki Nath Chowdhury, Tao Xiang, and Yi-Zhe Song. Doodle your 3D: From abstract freehand sketches to precise 3D shapes. In *CVPR*, 2024. 1, 2

[4] Ankan Kumar Bhunia, Salman Khan, Hisham Cholakkal, Rao Muhammad Anwer, Fahad Shahbaz Khan, Jorma Laaksonen, and Michael Felsberg. DoodleFormer: Creative sketch drawing with transformers. In *ECCV*, 2022. 3

[5] Rui Chen, Jianfeng Zhang, Yixun Liang, Guan Luo, Weiyu Li, Jiarui Liu, Xiu Li, Xiaoxiao Long, Jiashi Feng, and Ping Tan. Dora: Sampling and benchmarking for 3D shape variational auto-encoders. In *CVPR*, 2025. 4

[6] Tianrun Chen, Chenglong Fu, Ying Zang, Lanyun Zhu, Jia Zhang, Papa Mao, and Lingyun Sun. Deep3dsketch+: Rapid 3D modeling from single free-hand sketches. In *International Conference on Multimedia Modeling*. Springer, 2023. 2, 7

[7] Tianrun Chen, Chenglong Fu, Lanyun Zhu, Papa Mao, Jia Zhang, Ying Zang, and Lingyun Sun. Deep3dsketch: 3D modeling from free-hand sketches with view-and structural-aware adversarial training. *arXiv:2312.04435*, 2023. 2

[8] Tianrun Chen, Chaotao Ding, Shangzhan Zhang, Chunan Yu, Ying Zang, Zejian Li, Sida Peng, and Lingyun Sun. Rapid 3D model generation with intuitive 3D input. In *CVPR*, 2024. 1, 2, 3

[9] Yiwen Chen, Chi Zhang, Xiaofeng Yang, Zhongang Cai, Gang Yu, Lei Yang, and Guosheng Lin. IT3D: Improved text-to-3D generation with explicit view synthesis. In *AAAI*, 2024. 2

[10] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *CVPR*, 2019. 2

[11] Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tulyakov, Alexander G Schwing, and Liang-Yan Gui. SDFusion: Multimodal 3D shape completion, reconstruction, and generation. In *CVPR*, 2023. 2, 6

[12] Zezhou Cheng, Menglei Chai, Jian Ren, Hsin-Ying Lee, Kyle Olszewski, Zeng Huang, Subhransu Maji, and Sergey Tulyakov. Cross-modal 3D shape generation and manipulation. In *ECCV*, 2022. 2

[13] Changwoon Choi, Jaeah Lee, Jaesik Park, and Young Min Kim. 3Doodle: Compact abstraction of objects with 3D strokes. *ACM Transactions on Graphics (TOG)*, 2024. 3

[14] Johanna Delanoy, Mathieu Aubry, Phillip Isola, Alexei A Efros, and Adrien Bousseau. 3D sketching using multi-view deep volumetric prediction. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 2018. 2

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NACL*, 2019. 2, 4

[16] Rao Fu, Xiao Zhan, Yiwen Chen, Daniel Ritchie, and Srinath Sridhar. Shapecrafter: A recursive text-conditioned 3D shape generation model. *NeurIPS*, 2022. 1, 2

[17] Songen Gu, Haoxuan Song, Binjie Liu, Qian Yu, Sanyi Zhang, Haiyong Jiang, Jin Huang, and Feng Tian. VRsketch2Gaussian: 3D VR sketch guided 3D object generation with gaussian splatting. *arXiv:2503.12383*, 2025. 2, 3

[18] Benoit Guillard, Edoardo Remelli, Pierre Yvernay, and Pascal Fua. Sketch2mesh: Reconstructing and editing 3D shapes from sketches. In *ICCV*, 2021. 1, 2, 3

[19] David Ha and Douglas Eck. A neural representation of sketch drawings. *arXiv:1704.03477*, 2017. 3

[20] David Ha and Douglas Eck. A neural representation of sketch drawings. *arXiv:1704.03477*, 2017. 3

[21] Felix Hähnlein, Changjian Li, Niloy J Mitra, and Adrien Bousseau. Cad2sketch: Generating concept sketches from cad sequences. *ACM Transactions on Graphics (TOG)*, 41(6): 1–18, 2022. 2

[22] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. Teddy: a sketching interface for 3d freeform design. In *ACM SIGGRAPH 2006 Courses*, pages 11–es. 2006. 2

[23] Olga A Karpenko and John F Hughes. Smoothsketch: 3d free-form shapes from complex sketches. In *ACM SIGGRAPH 2006 Papers*, pages 589–598. 2006. 2

[24] Mohammad S Khan, Sankalp Sinha, Talha U Sheikh, Didier Stricker, Sk A Ali, and Muhammad Z Afzal. Text2CAD: Generating sequential cad designs from beginner-to-expert level text prompts. *NeurIPS*, 2024. 3

[25] Di Kong, Qiang Wang, and Yonggang Qi. A diffusion-refinement model for sketch-to-point modeling. In *ACCV*, 2022. 2

[26] Yunjin Lee, Lee Markosian, Seungyong Lee, and John F Hughes. Line drawings via abstracted shading. *ACM Transactions on Graphics (TOG)*, 26(3):18–es, 2007. 2

[27] Lei Li, Songyou Peng, Zehao Yu, Shaohui Liu, Rémi Pautrat, Xiaochuan Yin, and Marc Pollefeys. 3D neural edge reconstruction. In *CVPR*, 2024. 4

[28] Mengtian Li, Zhe Lin, Radomir Mech, Ersin Yumer, and Deva Ramanan. Photo-Sketching: Inferring contour drawings from images. In *WACV*, 2019. 3

[29] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3D: High-resolution text-to-3D content creation. In *CVPR*, 2023. 1, 2

[30] Hangyu Lin, Yanwei Fu, Xiangyang Xue, and Yu-Gang Jiang. Sketch-BERT: Learning sketch bidirectional encoder representation from transformers by self-supervised learning of sketch Gestalt. In *CVPR*, 2020. 3, 5, 8

[31] Chenxi Liu, Enrique Rosales, and Alla Sheffer. StrokeAggregator: Consolidating raw sketches into artist-intended curve drawings. *ACM Transactions on Graphics (TOG)*, 2018. 4

[32] Fang Liu, Xiaoming Deng, Yu-Kun Lai, Yong-Jin Liu, Cuixia Ma, and Hongan Wang. SketchGAN: Joint sketch completion and recognition with generative adversarial network. In *CVPR*, 2019. 3

[33] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3D mesh in 45 seconds without per-shape optimization. *NeurIPS*, 2023. 2

[34] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3D object. In *ICCV*, 2023. 2

[35] Yujia Liu, Anton Obukhov, Jan Dirk Wegner, and Konrad Schindler. Point2Cad: Reverse engineering cad models from 3D point clouds. In *CVPR*, 2024. 1

[36] Zhen Liu, Yao Feng, Michael J. Black, Derek Nowrouzezahrai, Liam Paull, and Weiyang Liu. Meshdiffusion: Score-based generative 3D mesh modeling. In *ICLR*, 2023. 2

[37] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*. 1998. 8

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[39] Zhaoliang Lun, Matheus Gadelha, Evangelos Kalogerakis, Subhransu Maji, and Rui Wang. 3D shape reconstruction from sketches via multi-view convolutional networks. In *3DV*. IEEE, 2017. 2

[40] Ling Luo, Yulia Gryaditskaya, Yongxin Yang, Tao Xiang, and Yi-Zhe Song. Fine-grained VR sketching: Dataset and insights. In *3DV*, 2021. 1, 2, 3, 6, 7

[41] Ling Luo, Pinaki Nath Chowdhury, Tao Xiang, Yi-Zhe Song, and Yulia Gryaditskaya. 3D VR sketch guided 3D shape prototyping and exploration. In *ICCV*, 2023. 1, 2, 5, 6, 8

[42] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3D point cloud generation. In *CVPR*, 2021. 2

[43] Matthew Matl. Pyrender. `https://github.com/mmatl/pyrender`, 2019. 8

[44] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3D shapes and textures. In *CVPR*, 2023. 2

[45] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 2021. 5

[46] Paritosh Mittal, Yen-Chi Cheng, Maneesh Singh, and Shubham Tulsiani. AutoSDF: Shape priors for 3D completion, reconstruction and generation. In *CVPR*, 2022. 2

[47] Gimin Nam, Mariem Khlifi, Andrew Rodriguez, Alberto Tono, Linqi Zhou, and Paul Guerrero. 3D-LDM: Neural implicit 3D shape generation with latent diffusion models. *arXiv:2212.00842*, 2022. 2

[48] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019. 2

[49] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3D using 2D diffusion. *arXiv:2209.14988*, 2022. 2

[50] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 2017. 3, 8

[51] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICLR*. PmLR, 2021. 3

[52] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 6

[53] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*. Springer. 6

[54] Aditya Sanghi, Hang Chu, Joseph G Lambourne, Ye Wang, Chin-Yi Cheng, Marco Fumero, and Kamal Rahimi Malekshan. Clip-forge: Towards zero-shot text-to-shape generation. In *CVPR*, 2022. 1

[55] Patsorn Sangkloy, Wittawat Jitkrittum, Diyi Yang, and James Hays. A sketch is worth a thousand words: Image retrieval with text and sketch. In *ECCV*. Springer, 2022. 1

[56] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. MVDream: Multi-view diffusion for 3D generation. *arXiv:2308.16512*, 2023. 2

[57] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 20121. 6

[58] Jifei Song, Kaiyue Pang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Learning to sketch with shortcut cycle consistency. In *CVPR*, 2018. 3

[59] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3D: High-fidelity 3D creation from a single image with diffusion prior. In *ICCV*, 2023. 2

[60] Maxim Tatarchenko, Stephan R Richter, René Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox. What do single-view 3D reconstruction networks learn? In *CVPR*, 2019. 6

[61] Xi Tian, Yong-Liang Yang, and Qi Wu. Shapescaffolder: Structure-aware 3D shape generation from text. In *ICCV*, 2023. 2

[62] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *NeurIPS*, 2017. 6

[63] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 3, 5

[64] Yael Vinker, Ehsan Pajouheshgar, Jessica Y Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. Clipasso: Semantically-aware object sketching. *ACM Transactions on Graphics (TOG)*, 2022. 3

[65] Yael Vinker, Yuval Alaluf, Daniel Cohen-Or, and Ariel Shamir. ClipAScene: Scene sketching with different types and levels of abstraction. In *CVPR*, 2023. 3

[66] Chuang Wang, Haitao Zhou, Ling Luo, and Qian Yu. ViewCraft3D: High-fidelity and view-consistent 3D vector graphics synthesis. *arXiv:2505.19492*, 2025. 3

[67] Jiayun Wang, Jierui Lin, Qian Yu, Runtao Liu, Yubei Chen, and Stella X Yu. 3D shape reconstruction from free-hand sketches. In *ECCV*. Springer, 2022. 2

[68] 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)*, 2021. 1

[69] Georges Winkenbach and David H Salesin. Computer-generated pen-and-ink illustration. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 91–100, 1994. 2

[70] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. *NeurIPS*, 2016. 2

[71] Rundi Wu, Xuelin Chen, Yixin Zhuang, and Baoquan Chen. Multimodal shape completion via conditional generative adversarial networks. In *ECCV*. Springer, 2020. 2

[72] Ximing Xing, Chuang Wang, Haitao Zhou, Jing Zhang, Qian Yu, and Dong Xu. DiffSketcher: Text guided vector sketch synthesis through latent diffusion models. *NeurIPS*, 2023. 3

[73] Xiang Xu, Joseph Lambourne, Pradeep Jayaraman, Zhengqing Wang, Karl Willis, and Yasutaka Furukawa. BrepGen: A B-rep generative diffusion model with structured latent geometry. *ACM Transactions on Graphics (TOG)*, 2024. 3

[74] Xingguang Yan, Liqiang Lin, Niloy J Mitra, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. ShapeFormer: Transformer-based shape completion via sparse representation. In *CVPR*, 2022. 2

[75] Qian Yu, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy Hospedales. Sketch-a-net that beats humans. *arXiv:1501.07873*, 2015. 3

[76] Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M Hospedales, and Chen-Change Loy. Sketch me that shoe. In *CVPR*, 2016. 1

[77] Ying Zang, Chaotao Ding, Tianrun Chen, Papa Mao, and Wenjun Hu. Deep3dsketch+\+: High-fidelity 3D modeling from single free-hand sketches. In *2023 IEEE International Conference on Systems, Man, and Cybernetic*. IEEE, 2023. 2

[78] Song-Hai Zhang, Yuan-Chen Guo, and Qing-Wen Gu. Sketch2model: View-aware 3D modeling from single free-hand sketches. In *CVPR*, 2021. 1, 2

[79] Yibo Zhang, Lihong Wang, Changqing Zou, Tieru Wu, and Rui Ma. Diff3DS: Generating view-consistent 3D sketch via differentiable curve rendering. In *ICLR*, 2024. 3

[80] Xinyang Zheng, Yang Liu, Pengshuai Wang, and Xin Tong. SDF-StyleGAN: Implicit SDF-based stylegan for 3D shape generation. In *Computer Graphics Forum*. Wiley Online Library, 2022. 2

[81] Xin-Yang Zheng, Hao Pan, Peng-Shuai Wang, Xin Tong, Yang Liu, and Heung-Yeung Shum. Locally attentional SDF diffusion for controllable 3D shape generation. *ACM Transactions on Graphics (ToG)*, 2023. 1, 2, 3, 6, 8

[82] Linqi Zhou, Yilun Du, and Jiajun Wu. 3D shape generation and completion through point-voxel diffusion. In *ICCV*, 2021. 2