SPACECONTROL: INTRODUCING TEST-TIME SPATIAL CONTROL TO 3D GENERATIVE MODELING

Anonymous authors
Paper under double-blind review

Figure 1: SPACECONTROL enables spatially controlled 3D asset generation using simple geometric primitives such as *superquadrics* (light blue) or other geometry (*e.g.*, meshes). Top: rapid asset generation. From quick 3D sketches and brief text prompts, we can generate high quality assets. Bottom: fine-grained editing, including adjusting a chair's backrest and adding armrests (left) or precisely controlling a sofa's dimensions and pillow arrangements (right).

ABSTRACT

Generative methods for 3D assets have recently achieved remarkable progress, yet providing intuitive and precise control over the object geometry remains a key challenge. Existing approaches predominantly rely on text or image prompts, which often fall short in geometric specificity: language can be ambiguous, and images are cumbersome to edit. In this work, we introduce SPACECONTROL, a training-free test-time method for explicit spatial control of 3D generation. Our approach accepts diverse geometric inputs, from coarse primitives to detailed meshes, and conditions a powerful pre-trained generative model without additional training. A controllable parameter lets users trade off between geometric fidelity and output realism. Extensive quantitative evaluation and user studies demonstrate that SPACECONTROL outperforms both training-based and optimization-based baselines in geometric faithfulness while preserving high visual quality. Finally, we present an interactive user interface that enables online editing of superquadrics for direct conversion into textured 3D assets, facilitating practical deployment in creative workflows.

1 INTRODUCTION

Generating 3D assets is a fundamental step in building virtual worlds, useful for gaming, simulation, virtual reality applications, and digital design. Recently the field of 3D object generation gained immense traction, and we are now able to create assets of previously unseen quality (Xiang et al.,

2025; Zhang et al., 2024; Vahdat et al., 2022; Gao et al., 2022; Wu et al., 2025; Siddiqui et al., 2024; Zhao et al., 2025; Chen et al., 2025). A persistent challenge, however, is *controllability*, *i.e.*, how users can effectively steer generation to align with desired shapes and appearances.

Current controllable 3D generation methods rely mainly on text or image conditioning. Text is accessible and flexible but inherently ambiguous and ill-suited for specifying precise geometry. Images provide stronger alignment with 3D structures but are cumbersome to edit and not intuitive for fine-grained adjustments. As a result, neither modality enables artists or designers to directly manipulate the geometry of generated objects. A more natural paradigm is to allow users to interact with the generative model in *3D space*, starting from coarse or abstract geometry and refining toward detailed assets.

Existing methods that introduce 3D geometric control fall into two categories: *training-based* and *guidance-based*. Training-based methods fine-tune existing generative models to support a specific form of geometric input, *e.g.* LION (Vahdat et al., 2022) for voxel conditioning, and Spice-E (Sella et al., 2024) for primitive or mesh conditioning. These methods provide controllability but require retraining, which reduces the original model's generalization capabilities. In contrast, guidance-based methods such as LatentNeRF (Metzer et al., 2023) and Coin3D (Dong et al., 2024) act solely at inference time without retraining, but usually involve substantial optimization overhead and constrain 3D structure only indirectly. Other works enrich existing 3D assets with geometric and appearance detail (Michel et al., 2022; Chen et al., 2023; Barda et al., 2025), yet they assume fine-grained input geometry, limiting usability in creative workflows where artists often begin with coarse sketches.

In this work, we present SPACECONTROL, a training-free method that injects explicit geometric control into Trellis (Xiang et al., 2025), a recent framework for text- or image-conditioned 3D generation, by directly encoding user-specified geometry into its latent space and using it as explicit guidance. Our method requires no additional training and enables controllable generation from diverse forms of geometry, ranging from simple primitives to detailed meshes.

We compare SPACECONTROL against both training-based (Sella et al., 2024) and guidance-based (Dong et al., 2024) approaches, as well as a stronger training-based variant of Spice-E adapted to Trellis. Remarkably, despite requiring no fine-tuning, SPACECONTROL achieves superior geometric faithfulness while preserving visual realism. We further provide a user interface that allows online editing of superquadrics and real-time generation of textured assets, supporting practical deployment in design workflows.

In summary, our contributions are the following:

- We introduce a training-free guidance method that conditions a powerful pre-trained generative model (Trellis) on user-defined geometry via latent space intervention, enabling geometry-aware generation without the need for costly fine-tuning.
- We conduct extensive evaluations, including a user study and quantitative analysis, showing that our method outperforms prior state-of-the-art methods for shape-conditioned 3D asset generation.
- We develop an interactive user interface that enables online editing of superquadrics and their real-time conversion into detailed, textured 3D assets, supporting practical deployment in creative workflows.

2 RELATED WORK

2.1 3D GENERATIVE MODELS

The field of 3D generation has experienced a rapid growth during the past few years both in terms of output modalities and controllability. Similar to the first image diffusion models (Ramesh et al., 2021), early applications of diffusion models for 3D generation (Nichol et al., 2022) were conducting the diffusion process in the original input space and were limited in the generated output type. More recent approaches (Vahdat et al., 2022; Jun & Nichol, 2023) started running the generation in a more compact latent space, leading to substantial improvements both in terms of quality and efficiency. To achieve an even increased efficiency, (Zhang et al., 2024; Xiang et al., 2025) have

started to disentangle the modeling of the structure from the appearance, leading to unprecedented high-quality generations. The separate modeling of geometry and appearance opens the door to explicit forms of spatially grounded conditioning, as done in our SPACECONTROL.

2.2 CONTROLLABLE GENERATIVE MODELS

Given a pretrained generative model, there are two main approaches to introduce a new control modality: (1) methods which *finetune* a part or the whole network to take new types of conditioning as input, and (2) *training-free* methods which condition the generation via inference-time guidance. In the last years many approaches have been developed to control the generation of image generative models, enabling conditioning in several forms as strokes, depth maps, and human poses. The same cannot be said for the field of 3D generation, which is still at his infancy.

CONTROLLING IMAGE GENERATIVE MODELS

A wide variety of methods have been proposed to introduce new control modalities to image generative models. Among works based on *finetuning*, we identify two main lines of research. On one side, there are works based on ControlNet (Zhang et al., 2023; Bhat et al., 2024) which add conditional control to a section of the network by introducing a trainable copy connected to the original via zero convolution. The key idea is to learn to control the original network without throwing information from the original training. On the other side, there are approaches which add additional layers for additional control of the network (Garibi et al., 2025; Hertz et al., 2022). Among *training-free* methods (Von Rütte et al., 2024; Meng et al., 2022; Sajnani et al., 2025), one closely related to our work is SDEdit (Meng et al., 2022) which uses stroke paintings to condition the generation of SDE-based generative models for images, by leveraging the denoising process of SDE-based generative models.

CONTROLLING 3D GENERATIVE MODELS

Only limited works have explored spatially grounded control of 3D generative models. On one side, approaches as LatentNERF (Metzer et al., 2023), Fantasia3D (Chen et al., 2023), and Instant3dit (Barda et al., 2025) leverage timely test-time optimization to achieve shape-conditioned novel view synthesis. On the other side, Spice-E (Sella et al., 2024) achieves the same goal by finetuning Shap-E (Jun & Nichol, 2023) separately on chairs, tables and airplanes from ShapeNet (Chang et al., 2015). These approaches attempt explicit spatial control, but nonetheless fall short of introducing a method that's as usable in unconstrained settings as introduced in their 2D counterparts. The former still requires long optimization times and use the geometric input to condition the generation of the 2D projections of the 3D objects, instead of directly conditioning in 3D. The latter needs class-specific fine-tuning which limits the applicability in unconstrained settings and does not allow to model the strength of the geometric control.

3 PRELIMINARIES

Before introducing our SPACECONTROL, we review the foundations on which it builds: rectified flow matching, the Trellis generative model, as well as superquadrics.

3.1 RECTIFIED FLOW MODELS

Rectified flow models use a linear interpolation forward (diffusion) process where for a specific time step $t \in [0, 1]$, the latent $\mathbf{z}_t$ can be expressed as $\mathbf{z}_t = (1-t)\mathbf{z}_0 + t\epsilon$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, I)$ and $\mathbf{z}_0$ is a clean sample from the target data distribution. The backward (denoising) process is represented by a time dependent velocity field $\mathbf{v}(\mathbf{z}_t, t) = \nabla_t \mathbf{z}_t$. In practice, starting from a noisy sample $\mathbf{z}_1$, we can obtain the denoised version $\mathbf{z}_0$ by discretizing the time interval $[0, 1]$ into T discrete steps, possibly not uniformly distributed, and recursively applying the equation

$$\mathbf{z}_{t(i+1)} = \mathbf{z}_{t(i)} - \mathbf{v}_\theta\big(\mathbf{z}_{t(i)}, t(i)\big)\big(t(i) - t(i+1)\big)\,, \tag{1}$$

where $i \in [1, T-1]$ and the vector field $\mathbf{v}_\theta(\cdot)$ is predicted for example by a Diffusion Transformer (Peebles & Xie, 2023) as in Trellis (Xiang et al., 2025).

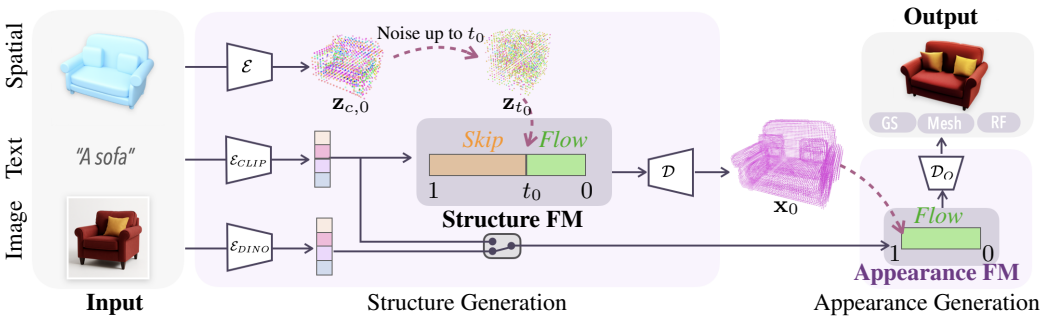


Figure 2: **Model Overview.** Given an input conditioning which includes a spatial control, a text prompt and an image (optional), SPACECONTROL produces realistic 3D assets. First the different conditioning are encoded in a latent space. Specifically, the spatial control is voxelized and encoded by Trellis' encoder $\mathcal{E}$, the text is encoded by a CLIP encoder $\mathcal{E}_{CLIP}$, and the image (if present) is encoded by a DINOv2 encoder $\mathcal{E}_{DINO}$. The obtained latents $\mathbf{z}_{0,c}$ are noised up to t_0 to obtain $\mathbf{z}_{t_0}$. From t_0 to $t = 0$, $\mathbf{z}_{t_0}$ are denoised by the *Structure Flow Model* (FM), guided by the text prompt features. The clean latents $\mathbf{z}_0$ are then fed into the decoder $\mathcal{D}$, which outputs the voxel grid $\mathbf{x}_0$. Then, the active voxels are augmented with point-wise noisy latent features, denoised by the *Appearance Flow Model* (FM), using either text or image conditioning. The clean latents can then be decoded into versatile output formats such as 3D gaussians (GS), radiance fields (RF), and meshes (M) via specific decoders $\mathcal{D}_O = \{\mathcal{D}_{GS}, \mathcal{D}_{RF}, \mathcal{D}_M\}$.

3.1.1 STEPS SCHEDULE

Time steps are initially defined as $t(\tau) = 1 - \tau/T$ for $\tau \in [0, T]$, and then rescaled by a factor λ:

$$t(\tau) = \frac{\lambda t(\tau)}{1 + (\lambda - 1)t(\tau)}. \tag{2}$$

Since t can be obtained from τ and vice versa, we will refer to either one interchangeably.

3.2 TRELLIS

Trellis (Xiang et al., 2025) is a recent 3D generative model which employs rectified flow models to generate 3D assets from either textual or image conditioning. Specifically, it consists of two separate steps of generations, where the first aims to generate the *structure*, while the second focus on the *appearance*.

3.2.1 STRUCTURE GENERATION

In the first step, a noisy latent variable $\mathbf{z}_1 \in \mathbb{R}^{16\times16\times16\times8}$ is sampled from $\mathcal{N}(\mathbf{0}, I)$ and denoised by a rectified flow model iteratively applying Eq. 1 using either image or text conditioning. Specifically, text conditions are encoded via a CLIP (Radford et al., 2021) text encoder, while image conditions are encoded via a DINOv2 (Oquab et al., 2024) encoder. The denoised latent $\mathbf{z}_0$ is then decoded by a decoder $\mathcal{D}$ to obtain a voxel grid $\mathbf{x} \in \{0, 1\}^{64\times64\times64}$, which encodes the spatial structure of the 3D asset. Notice that the decoder $\mathcal{D}$ is pretrained jointly with an associated encoder $\mathcal{E}$, not explicitly used in the Trellis pipeline.

3.2.2 APPEARANCE GENERATION

In the second step, the L active voxels are augmented with point-wise noisy latent features $\mathbf{s}_1 \in \mathbb{R}^{L\times8}$ sampled from $\mathcal{N}(\mathbf{0}, I)$, denoised by a second flow model, using either text or image conditioning. The clean latents $\mathbf{s}_0 \in \mathbb{R}^{L\times8}$ can then be decoded into versatile formats such as 3D gaussians (GS), radiance fields (RF), and meshes (M) via specific decoders $\mathcal{D}_O = \{\mathcal{D}_{GS}, \mathcal{D}_{RF}, \mathcal{D}_M\}$.

3.3 SUPERQUADRICS

Superquadrics (Barr, 1981) provide a compact parametric family of shapes capable of representing diverse geometries. A canonical superquadric is defined by five parameters: scales (s_x, s_y, s_z) and

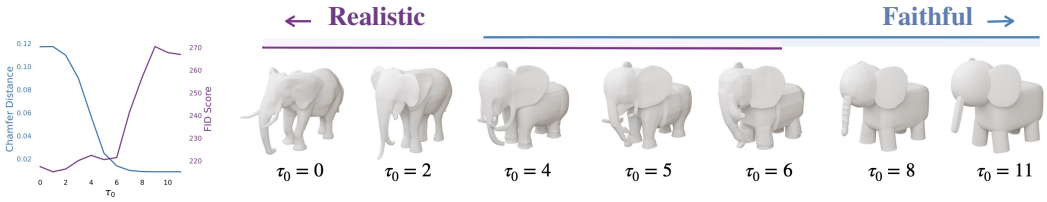


Figure 3: **Realism-faithfulness tradeoff.** The hyperparameter τ_0 allows a smooth control over the strength of the control. In the left figure we show how variations of τ_0 affects the generations quantitatively in terms of Chamfer distance to the spatial control (lower means more *faithful*) and of FID score (lower means more *realistic*). In the right figure we show it qualitatively, visualizing how higher values of τ_0 lead to assets whose geometry looks even more similar to the control. For conciseness we only show the untextured geometry.

exponents (ϵ_1, ϵ_2). With parametric coordinates (η, ω) we can define their surface as:

$$s(\eta, \omega) = \begin{bmatrix} s_x \cos(\eta)^{\epsilon_1} \cos(\omega)^{\epsilon_2} \\ s_y \cos(\eta)^{\epsilon_1} \sin(\omega)^{\epsilon_2} \\ s_z \sin(\eta)^{\epsilon_1} \end{bmatrix}. \tag{3}$$

Extending to world coordinates requires 6 additional pose parameters (3 translation, 3 rotation), giving 11 parameters in total. Their compactness makes them well-suited as spatial control primitives.

4 METHOD

We start introducing our problem setup in Sec. 4.1. We present our approach in Sec. 4.2, and discuss how we achieve a flexible control over the strength of the spatial control in Sec. 4.3.

4.1 SETUP

To introduce spatial control in the generation of 3D models the user needs to provide a geometric conditioning, together with a text prompt. Our goal is to produce 3D assets with two desiderata:

- **Faithfulness**: the generated asset should be aligned with the control geometry.
- **Realism**: the generated asset should retain the quality of the original model.

4.2 APPROACH

In this section we introduce SPACECONTROL and describe how it can perform guided generation of 3D assets by introducing spatial guidance to a pretrained Trellis model. As our control strategy differs from the first to the second stage of generation, we explain how we guide the former in Sec. 4.2.1 and the latter in Sec. 4.2.2.

4.2.1 STRUCTURE GENERATION

To control the first step of generation given an explicit control geometry we employ a similar framework to SEdit (Meng et al., 2022), where instead of using *strokes* to guide the generation of *2D images*, we use either coarse or detailed 3D geometry to guide the generation of *3D assets*. Specifically, given a user-specified 3D geometry, we voxelize it to obtain $\mathbf{x}_c \in \{0,1\}^{64\times64\times64}$ and feed $\mathbf{x}_c$ into the pretrained encoder $\mathcal{E}$ to obtain $\mathbf{z}_{c,0} \in \mathbb{R}^{16\times16\times16\times8}$. Then given a specific time step $t_0 \in [0,1]$ we noise up the latents $\mathbf{z}_{c,0}$ to that specific step via the rectified flows forward equation as:

$$\mathbf{z}_{t_0} = t_0 \mathbf{z}_1 + (1 - t_0)\, \mathbf{z}_{c,0}\,, \tag{4}$$

where $\mathbf{z}_1 \sim \mathcal{N}(\mathbf{0}, I)$. Given $\mathbf{z}_{t_0}$, $\mathbf{z}_0$ can then obtained by iteratively applying Eq. 1 starting from t_0 and employing the by the original *Structure Flow Model*. We note that this process does not require

any need of architectural changes nor training. We guide the generation with additional textual prompt, which is helpful to disambiguate the semantic of the object. As in the standard setting, the denoised latent $\mathbf{z}_0$ is then decoded into a final geometric structure $\mathbf{x}_0 \in \{0,1\}^{64\times64\times64}$ by $\mathcal{D}$.

4.2.2 Appearance generation

Given the geometric structure generated in the first stage, we then employ either text or image conditioning to guide the generation of its appearance, by first expanding the active voxels with point-wise noisy latent features and then denoising them using the *Appearance Flow Model*. Notice that, even if the structure generation is always conditioned on text, image conditioning can still be used in to guide the appearance generation, allowing for finer control over the visual details (see Fig. 6a and Appendix).

4.3 Controlling the strength of spatial control

The strength of spatial control can be tuned through the parameter τ_0. For lower values of τ_0, the latent z_{t_0} is initialized closer to the noise z_1 than to the control signal $z_{c,0}$, leading the model to perform more denoising steps. This favors samples that follow the data distribution of the original Trellis, producing outputs that are generally more realistic but less faithful to the spatial conditioning. In contrast, higher values of τ_0 bias z_{t_0} towards $z_{c,0}$, effectively skipping earlier denoising steps and preserving more of the injected spatial structure, albeit sometimes at the expense of realism.

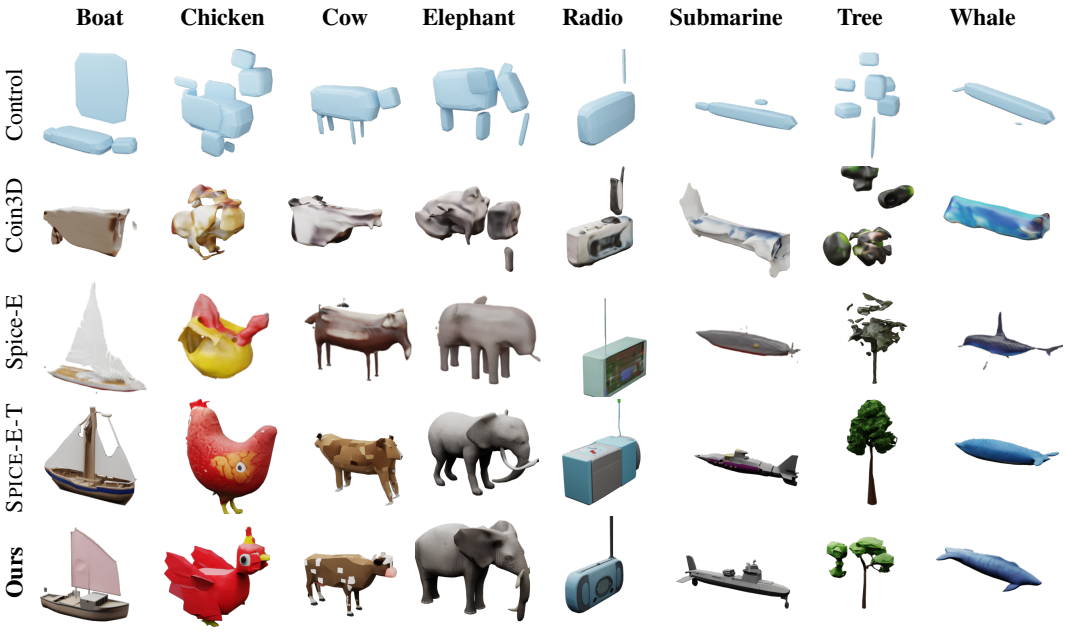


Figure 4: **Qualitative Comparison of Spatially Conditioned Generation.** We show generations obtained conditioning our SpaceControl and baselines on text prompts and superquadrics from the Toys4K dataset. While other methods either fails to follow the conditioning (*e.g.*, the antenna from the radio generated by Spice-E is wrongly placed) or to generate visually appealing 3D assets (*e.g.*, the chicken generated by Spice-E-T exhibits anatomically incorrect body part placements), SpaceControl exhibits a good balance between realism and faithfulness.

5 Experiments

5.1 Comparing with State-of-the-art Methods

Tasks We evaluate the capabilities of our SpaceControl when the spatial condition is provided as (1) *coarse* and (2) *detailed* geometry. In the former case we employ simple geometric primitives, in the latter detailed object meshes.

Table 1: **Comparison with Baselines.** The evaluation metrics are L2 *Chamfer Distance* (CD) and *Fréchet Inception Distance* (FID). CD quantifies alignment with spatial control, while FID assesses realism. Results for SPACECONTROL are reported at $\tau_0 = 6$. CD scores are multiplied by 10^3. † indicates methods fine-tuned on *chair* and *table*. Trellis (Xiang et al., 2025) (model: txt-DiT-XL) does not offer spatial guidance, and is shown for reference only.

Method	Toys4K				Chair				Table			
	CD↓	CLIP-I↑	FID↓	P-FID↓	CD↓	CLIP-I↑	FID↓	P-FID↓	CD↓	CLIP-I↑	FID↓	P-FID↓
TRELLIS	117	0.33	217	78.60	14.7	0.31	129	40.82	19.7	0.30	132	49.40
Geometric Primitives												
Coin3D	54.4	0.21	231	102.0	18.5	0.25	218	47.54	28.82	0.22	245	71.58
Spice-E†	65.9	0.29	233	66.52	7.66	0.29	166	38.66	10.3	0.29	148	78.85
SPICE-E-T†	39.1	**0.32**	223	**53.51**	5.92	**0.31**	**135**	39.22	4.73	**0.30**	**122**	47.36
SPACECONTROL (Ours)	**14.0**	**0.32**	**221**	81.3	**0.98**	0.30	146	**34.06**	**3.72**	0.29	157	**46.28**
Meshes												
Coin3D	77.8	0.04	293	182.5	14.6	0.01	308	111.0	20.4	0.01	224	178.2
Spice-E (stylization)	7.40	0.30	224	81.21	6.37	0.30	152	41.51	28.2	0.29	132	58.01
SPICE-E-T†	23.3	**0.32**	**222**	90.99	22.7	**0.31**	**132**	39.70	7.59	**0.30**	**116**	46.76
SPACECONTROL (Ours)	**4.89**	0.29	244	**72.47**	**0.66**	0.29	137	**30.96**	**0.48**	0.28	130	**42.33**

Baselines We compare SPACECONTROL to state-of-the-art *training-based* and *guidance-based* baselines. As *training-based* baseline we compare to *Spice-E* (Sella et al., 2024), which fine-tunes Shap-E (Jun & Nichol, 2023) to support *cuboid* primitives as spatial guidance for 3D object generation. Since Spice-E is based on the *Shap-E* model (Jun & Nichol, 2023), to allow a fairer comparison we implement its correspondent for Trellis (Xiang et al., 2025), which we will refer to as SPICE-E-T. We provide more details on its implementation and training in the Appendix. Note that *Spice-E* provides a separate checkpoint for shape stylization, which is used to evaluate the method on mesh conditioning, as it lead to better results. As *guidance-based* baseline we compare to Coin3D (Dong et al., 2024), which uses the shape-guidance to generate consistent multiple views of the desired 3D asset and then interpolate them in 3D by training a NeRF (Mildenhall et al., 2020) for 2000 iterations, and finally extract a mesh using.

Datasets To evaluate how different approaches handle geometric conditioning, we create a dataset of objects which contains the original mesh, a decomposition of it into geometric primitives, and a textual description of the asset. We use the mesh to evaluate methods on mesh-conditioned generation and geometric primitives to evaluate on shape-conditioned generation. Moreover, to evaluate both *generation* and *generalization* capabilities, we use objects of two ShapeNet (Chang et al., 2015) categories (chairs and tables) that Spice-E was explicitly trained on together with objects from the Toys4K (Stojanov et al., 2021) dataset, unseen by all methods during training. We use SuperDec (Fedele et al., 2025) to obtain the decomposition of the 3D assets into superquadrics and Gemini on rendered views to obtain a textual description of the assets from ShapeNet (Chang et al., 2015). For objects from Toys4k we use the textual description from Xiang et al. (2025).

Metrics Our experiments aim to evaluate both the *faithfulness* to the spatial and textual control and the *realism* of the generated assets. Faithfulness to the spatial control is quantified using the L2 *Chamfer Distance* (CD) between vertices sampled from the input superquadric primitives and the generated mesh decoded by $\mathcal{D}_M$. Faithfulness to the textual control is quantified with the CLIP similarity (CLIP-I) between the renderings of generated assets and the textual prompts. Realism is evaluated for texture via the *Fréchet Inception Distance* (FID) (Heusel et al., 2017) on image renderings and for geometry, via the P-FID (Nichol et al., 2022), the point cloud analog for FID. To measure the FID on image rendering we measure the distance between the inception features extracted from the original image renderings of the datasets and the generated ones. To measure the P-FID of the generated meshes we measure the distance between the PointNet++ (Qi et al., 2017) features of the generated and original object meshes.

Results Quantitative results are reported in Table 1, while qualitative results are shown in Figure 4. Both Spice-E and SPICE-E-T perform well on *chairs* and *tables* but struggle to faithfully generate objects that they were not fine-tuned on (Toys4K). SPACECONTROL significantly outperforms the baselines in all experiments in terms of Chamfer Distance (CD) to the spatial control, while achieving comparable CLIP-I, FID, and P-FID scores. For completeness, we also report scores for the text-conditioned Trellis using the DiT-XL backbone, which is also the base model used in our

Table 2: **Analysis of τ_0.** The evaluation metrics are L2 *Chamfer Distance* (CD) and *Fréchet Inception Distance* (FID). CD quantifies alignment with spatial control, while FID assesses realism. CD scores are scaled by 10^3. We show scores for spatial control given as geometric primitives (P) and meshes (M).

	Toys4K								Chair								Table							
	CD ↓		CLIP-I ↑		FID ↓		P-FID ↓		CD ↓		CLIP-I ↑		FID ↓		P-FID ↓		CD ↓		CLIP-I ↑		FID ↓		P-FID ↓	
τ_0	P	M	P	M	P	M	P	M	P	M	P	M	P	M	P	M	P	M	P	M	P	M	P	M
0	117	75.4	**0.33**	**0.29**	217	254.9	**78.6**	79.4	14.7	30.6	**0.31**	**0.29**	**129**	**133.7**	40.8	39.9	19.7	49.21	**0.30**	**0.28**	**132**	137.5	49.40	49.3
2	110	65.5	**0.33**	**0.29**	**216**	256.9	79.1	82.7	14.1	30.0	**0.31**	**0.29**	131	136.7	41.2	41.5	18.5	43.51	**0.30**	**0.28**	**132**	134.7	51.97	**41.5**
4	56.8	32.4	0.32	**0.29**	222	252.8	84.1	83.9	7.3	13.9	**0.31**	**0.29**	137	141.1	34.1	31.9	6.33	2.68	**0.30**	**0.28**	135	133.5	51.79	45.8
6	14.0	4.89	0.32	**0.29**	221	244.9	81.3	**72.5**	0.98	0.66	0.30	**0.29**	146	136.6	**34.0**	31.0	3.72	0.48	0.29	**0.28**	157	131.0	**46.28**	42.3
8	9.04	1.57	0.29	**0.29**	257	241.3	94.0	77.0	0.27	0.28	0.30	0.28	156	134.3	37.1	**29.2**	3.29	**0.19**	0.29	**0.28**	175	127.3	50.16	43.2
10	**8.85**	**1.84**	0.27	**0.29**	268	**209.3**	101	74.9	**0.22**	**0.26**	0.30	0.28	160	134.0	36.5	30.1	**3.26**	**0.19**	0.29	0.29	181	**125.9**	50.74	42.6

SPACECONTROL. Note that for the sake of simplicity in Tab. 1 we only report results of SPACECONTROL with $\tau_0 = 6$. However, τ_0 can be chosen freely by the user, depending on the desired strength of conditioning. For completeness, we report results for different values of τ_0 in Tab. 2. We can see that by increasing the value of τ_0 and thus strength of the spatial conditioning, we obtain generations which align more closely to the input spatial control.

User Study. To validate the numerical results, we conduct a user study (Fig. 5) involving 52 volunteers, each one evaluating on average 20 randomly selected samples. Participants were asked to compare pairs of generated objects, voting which one was more faithful to the input control shape, which model looked more realistic, and which one they liked overall better (see appendix for more details). The study is performed on the same datasets discussed above, *i.e.*on ShapeNet (Chang et al., 2015) and Toys4k (Stojanov et al., 2021). We compare our SPACECONTROL to the Spice-E and Spice-E-T baselines. We observe that our SPACECONTROL is always the preferred method both in terms of overall appearance and alignment to the input spatial control.

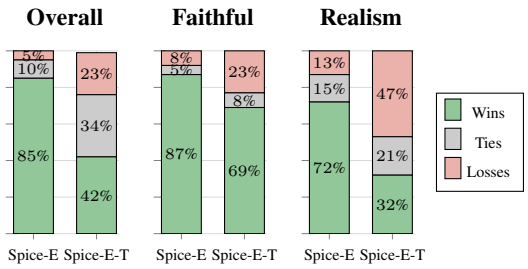


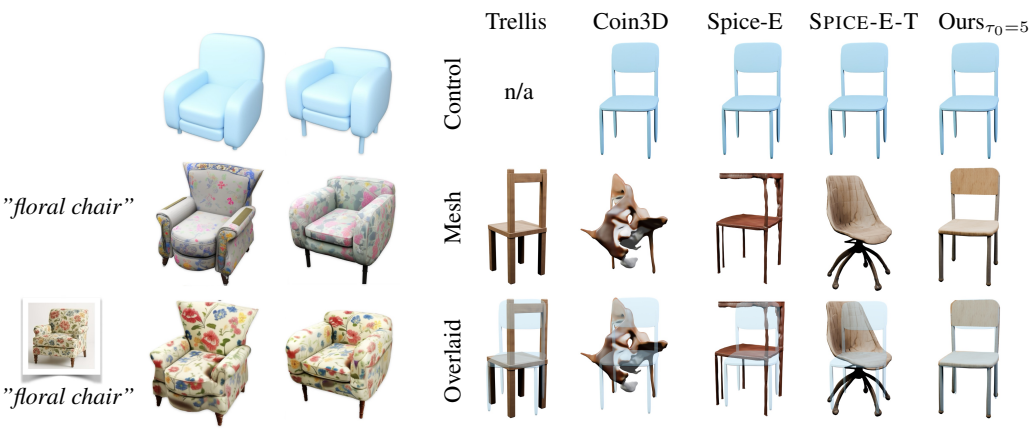
Figure 5: **User Study Results.** The bar plots present the proportion of favorable comparisons achieved by our SPACECONTROL against the baselines on overall appearance, faithfulness to spatial control, and realism, respectively.

5.2 QUALITATIVE RESULTS

Besides Figures 1, 4, and 6, we provide additional qualitative results for object editing in the Appendix, visualizing outputs of different methods conditioned on both coarse and detailed input controls. In general, training-based methods struggle to generate objects in specific poses, whereas SPACECONTROL consistently produces plausible results. For example, other methods generate a cow with two heads (Spice-E and Spice-E-T), an elephant with an eye on its back (Spice-E), or shapes that fail to strictly follow the spatial conditioning or exhibit low quality (Coin3D).

5.3 ANALYSIS EXPERIMENTS

The Effect of the Control Parameter τ_0. While existing methods for 3D spatial conditioning do not provide a way to control its strength, our SPACECONTROL enables flexible interpolation between different levels of adherence. In this section, we evaluate how the parameter τ_0 governs the trade-off between fidelity to the spatial control signal and the realism of the generated asset. Quantitative results are reported in Table 2, using the same metrics and datasets as in Table 1. We further present qualitative results in Fig. 3 and in the Appendix, showing how varying the conditioning strength produces different outcomes. Adjusting τ_0 allows users to regulate this trade-off according to their preferences, balancing higher shape quality against stronger adherence to the spatial guidance. Additionally, the plot in Figure 3 *(left)* illustrates this trade-off on Toys4K, indicating that $\tau_0 \in [4, 6]$ generally provides a good compromise between spatial adherence and shape quality.



(a) **Image conditioning.** Given the two different spatial controls shown in the*first row*, we show objects generated by our SPACECONTROL without *(second row)* and with *(third row)* image conditioning.

(b) **Spatial alignment.** We show how different methods align the generated 3D asset with the input condition. In the first row we show the input control, in the second the generated asset and in the third, we overlay the two. All the generations use the same prompt *"A wooden chair."*.

Figure 6: **Image conditioning and fine-grained alignment.** We show analysis experiments on the role of image conditioning (*left*) and on fine-grained spatial alignment (*right*).

The Role of Image Conditioning. SPACECONTROL supports multi-modal control for 3D asset generation by combining spatial guidance via superquadrics with natural language and optional image conditioning. While the model can synthesize assets using only superquadrics and textual prompts, images are particularly useful for maintaining visual consistency during object edits, as shown in Figure 6a and in the Appendix. As we only use image prompts in the *Appearance Flow Model* of Trellis, they primarily affect texture, with only minor influence on geometry. While this capability originates from the pre-trained Trellis, SPACECONTROL enables its practical use for cross-modal texture transfer, effectively performing style transfer from 2D images to generated 3D shapes.

Spatial Alignment. We believe that a key advantage of a training-free approach that performs conditioning directly in 3D space is its ability to achieve fine-grained spatial control. In this section, we provide an example where the conditioning shapes are not aligned with axis-oriented rotations. As shown in Fig. 6b, our method is the only one that perfectly aligns with the input conditioning while preserving the quality of the generated mesh. Additional results are provided in the Appendix.

6 DISCUSSION AND CONCLUSION

In summary, our approach introduces the first training-free method that by operating directly in the 3D space is able to spatially condition the generation of high quality assets. Through extensive evaluations and a practical interface, we demonstrate both the effectiveness and usability of our method in real-world creative workflows.

Limitations and future work. While SPACECONTROL enables flexible spatial control via a tunable adherence parameter τ_0, this parameter is currently selected manually. Although this supports user-driven control over the realism–faithfulness tradeoff, it complicates automated generation of diverse, high-quality assets without per-instance tuning. Additionally, our current formulation enforces a uniform adherence level across the entire object. Future work could explore part-aware control, allowing users to specify which regions should closely follow the input structure and which can deviate more freely to support creative variation.

Reproducibility statement. Our approach builds on the open-source Trellis model (Xiang et al., 2025), and our experiments use open-source datasets, namely ShapeNet (Chang et al., 2015) and Toys4k (Stojanov et al., 2021). All experiments are fully reproducible, and upon acceptance, we will release our code to facilitate replication of our method and results.

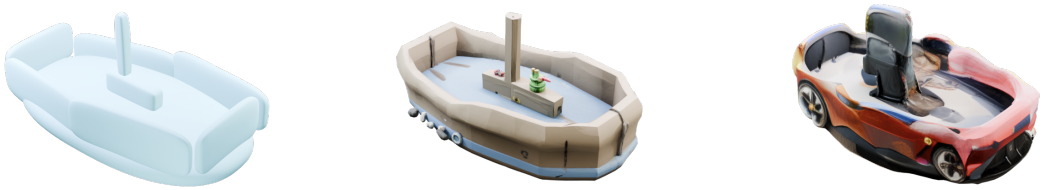
7 Additional Rebuttal Results

7.1 Local Control

Without *local* semantic conditioning. **With** *local* semantic conditioning.

Figure 7: **Local semantic control.** From left to right we show: the input geometric control, the 3D asset generated by globally conditioning on "*A white chair.*", the 3D asset generated by conditioning globally on "*A white chair.*" and locally (on the superquadric highlighted in red) on "*A read seat.*".

7.2 Semantically Contradictory Conditionings



Geometric guidance *"A boat."* *"A car."*

Figure 8: **Contrasting conditioning.** We use a coarse geometric sketch of a boat (*left*) as geometric control and pair it with two different textual prompts: *"A boat."* (*middle*) and *"A car."* (*right*). When the prompts align, SpaceControl produces a coherent result. When they conflict, the model injects car-like appearance cues (e.g. wheels) while preserving the underlying boat geometry.

7.3 Tapering and Bending of Superquadrics

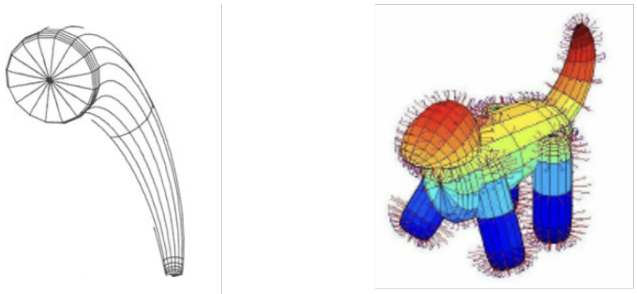


Figure 9: **Tapering and bending of superquadrics.** A superquadric with tapering and bending transformations (*left*, from Jaklic et al. (2000)) and an animal composed by superquadrics with bending and taperings (*right*, from Pelossof et al. (2004)).

REFERENCES

Amir Barda, Matheus Gadelha, Vladimir G Kim, Noam Aigerman, Amit H Bermano, and Thibault Groueix. Instant3dit: Multiview inpainting for fast editing of 3d objects. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2, 3

Alan H. Barr. Superquadrics and Angle-Preserving Transformations. *IEEE Computer Graphics and Applications*, 1981. 4

Shariq Farooq Bhat, Niloy Mitra, and Peter Wonka. Loosecontrol: Lifting controlnet for generalized depth conditioning. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024. 3

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. ShapeNet: An Information-rich 3D Model Repository. *arXiv preprint arXiv:1512.03012*, 2015. 3, 7, 8, 9

Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3D: Disentangling Geometry and Appearance for High-quality Text-to-3D Content Creation. In *International Conference on Computer Vision (ICCV)*, 2023. 2, 3

Zhaoxi Chen, Jiaxiang Tang, Yuhao Dong, Ziang Cao, Fangzhou Hong, Yushi Lan, Tengfei Wang, Haozhe Xie, Tong Wu, Shunsuke Saito, et al. 3DTopia-XL: Scaling High-quality 3D Asset Generation via Primitive Diffusion. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2

Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. ABO: Dataset and Benchmarks for Real-world 3D Object Understanding. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 18

Wenqi Dong, Bangbang Yang, Lin Ma, Xiao Liu, Liyuan Cui, Hujun Bao, Yuewen Ma, and Zhaopeng Cui. Coin3D: Controllable and Interactive 3D Assets Generation with Proxy-Guided Conditioning. In *ACM SIGGRAPH 2024 Conference Papers*, 2024. 2, 7

Elisabetta Fedele, Boyang Sun, Leonidas Guibas, Marc Pollefeys, and Francis Engelmann. SuperDec: 3D Scene Decomposition with Superquadric Primitives. In *International Conference on Computer Vision (ICCV)*, 2025. 7, 18

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. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 2

Daniel Garibi, Shahar Yadin, Roni Paiss, Omer Tov, Shiran Zada, Ariel Ephrat, Tomer Michaeli, Inbar Mosseri, and Tali Dekel. Tokenverse: Versatile multi-concept personalization in token modulation space. *ACM Transactions On Graphics (TOG)*, 2025. 3

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv*, 2022. 3

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-scale Update Rule Converge to a Local Nash Equilibrium. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2017. 7

Ales Jaklic, Ales Leonardis, and Franc Solina. *Segmentation and recovery of superquadrics*, volume 20. Springer Science & Business Media, 2000. 10

Heewoo Jun and Alex Nichol. Shap-E: Generating Conditional 3D Implicit Functions. *arXiv preprint arXiv:2305.02463*, 2023. 2, 3, 7

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations. In *International Conference on Learning Representations (ICLR)*, 2022. 3, 5

Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-NeRF for Shape-guided Generation of 3D Shapes and Textures. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 3

Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2Mesh: Text-driven Neural Stylization for Meshes. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

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. In *European Conference on Computer Vision (ECCV)*, 2020. 7

Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-E: A System for Generating 3D Point Clouds from Complex Prompts. *arXiv preprint arXiv:2212.08751*, 2022. 2, 7

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research Journal*, 2024. 4

William Peebles and Saining Xie. Scalable Diffusion Models With Transformers. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

Raphael Pelossof, Andrew Miller, Peter Allen, and Tony Jebara. An svm learning approach to robotic grasping. In *International Conference on Robotics and Automation (ICRA)*, 2004. 10

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2017. 7

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 *International Conference on Machine Learning (ICML)*, 2021. 4

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-Shot Text-to-image Generation. In *International Conference on Machine Learning (ICML)*, 2021. 2

Rahul Sajnani, Jeroen Vanbaar, Jie Min, Kapil Katyal, and Srinath Sridhar. Geodiffuser: Geometry-based image editing with diffusion models. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025. 3

Etai Sella, Gal Fiebelman, Noam Atia, and Hadar Averbuch-Elor. Spic-E: Structural Priors in 3D Diffusion Models using Cross Entity Attention. *ACM SIGGRAPH Conference Papers*, 2024. 2, 3, 7

Yawar Siddiqui, Tom Monnier, Filippos Kokkinos, Mahendra Kariya, Yanir Kleiman, Emilien Garreau, Oran Gafni, Natalia Neverova, Andrea Vedaldi, Roman Shapovalov, et al. Meta 3D AssetGen: Text-to-mesh Generation with High-quality Geometry, Texture, and BPR Materials. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2024. 2

Stefan Stojanov, Anh Thai, and James M Rehg. Using Shape to Categorize: Low-shot Learning with an Explicit Shape Bias. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 7, 8, 9

Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, Karsten Kreis, et al. LION: Latent Point Diffusion Models for 3D Shape Generation. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 2

Dimitri Von Rütte, Elisabetta Fedele, Jonathan Thomm, and Lukas Wolf. Fabric: Personalizing diffusion models with iterative feedback. In *European Conference on Computer Vision (ECCV)*, 2024. 3

Tianhao Wu, Chuanxia Zheng, Frank Guan, Andrea Vedaldi, and Tat-Jen Cham. Amodal3R: Amodal 3D Reconstruction from Occluded 2D Images. *arXiv preprint arXiv:2503.13439*, 2025. 2

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3D Latents for Scalable and Versatile 3D Generation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 1, 2, 3, 4, 7, 9

Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. CLAY: A Controllable Large-scale Generative Model for Creating High-quality 3D Assets. *ACM Transactions on Graphics (TOG)*, 2024. 2

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding Conditional Control to Text-to-Image Diffusion Models. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, et al. Hunyuan3d 2.0: Scaling Diffusion Models for High Resolution Textured 3D Assets Generation. *arXiv preprint arXiv:2501.12202*, 2025. 2

A Additional Results

A.1 Fine-grained spatial editing

In this section we provide additional results which show how the generations from our SpaceControl are influenced by the change of the spatial control. We show results in pairs where the textual and/or image prompts are kept fixed. We notice that by providing additional image control, we are able to preserve the texture between different generations.

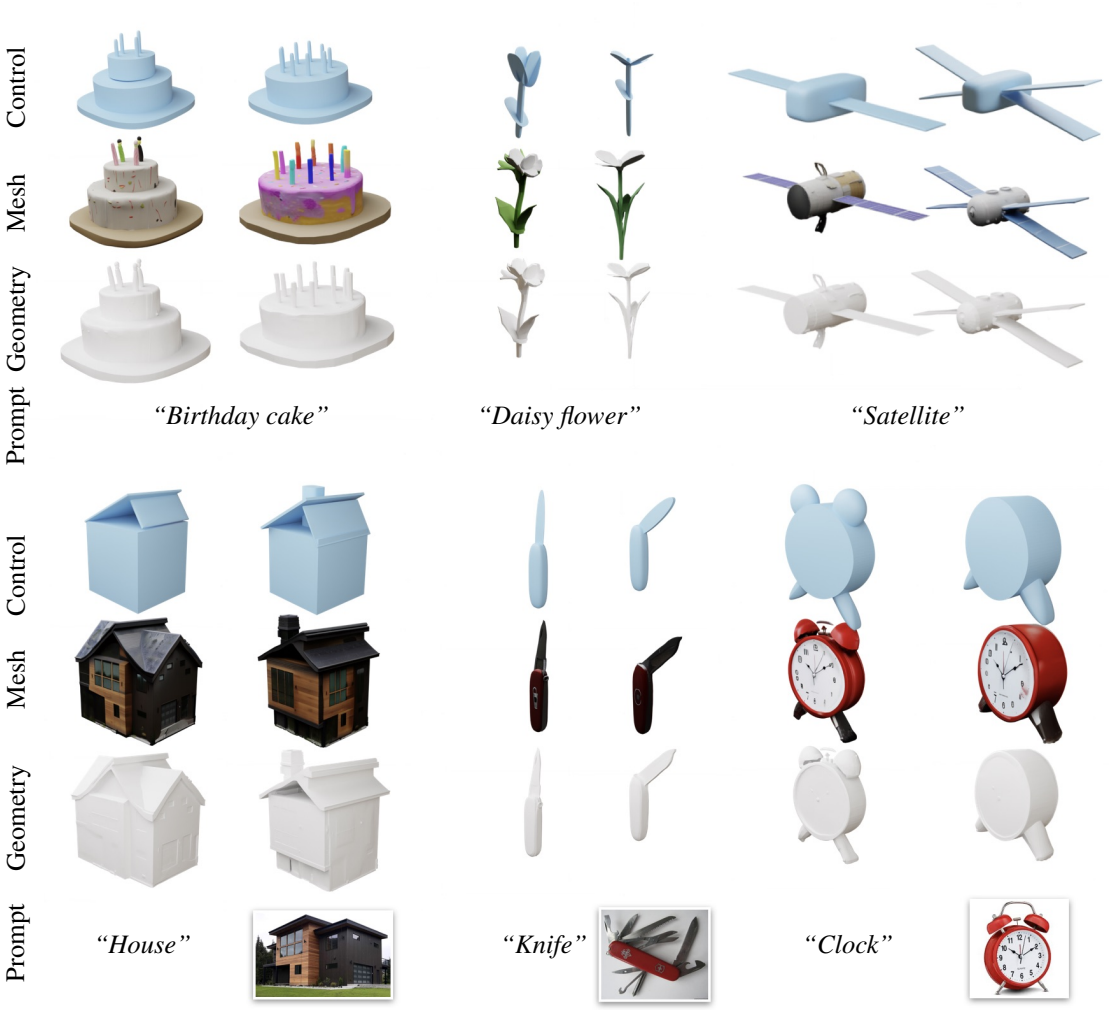


Figure 10: **Fine-grained spatial editing with superquadrics.** Superquadrics offer fine-grained spatial control that is useful not only for generating a wide variety of 3D assets, but also for editing them. They enable intuitive and localized modifications of 3D shapes, in a more direct manner than text- or image-only generative models in practicality. In addition to natural language prompts (*top*), SpaceControl supports image conditioned generation (*bottom*), enabling consistent visual appearance across edits.

A.2 COARSE AND FINE-GRAINED SPATIAL CONTROL WITH SUPERQUADRICS

In this section, we provide additional results generated with different control strengths. Here the hyperparameter is chosen so that we were satisfied with the final result. Superquadrics prove to be an effective tool to provide both coarse and fine-grained control to the 3D generation. By combining the expressivity of superquadrics with the flexible control strength offered by our SPACECONTROL, users can condition the generation by either carefully designing geometric details or only drafting the spatial setting of the desired output.

Figure 11: **Coarse and fine-grained control with superquadrics.** Superquadrics offer both fine-grained spatial control when used to sculpt precise geometry (*motorbike, staircase, helicopter*) and coarse control, when only used to draft a 3D sketch (*duck, drumkit*).

A.3 FINE-GRAINED ALIGNMENT WITH STATE-OF-THE-ART METHODS

In Fig. 12 we show the results for the same experiment provided in the main paper, but with different control strengths.

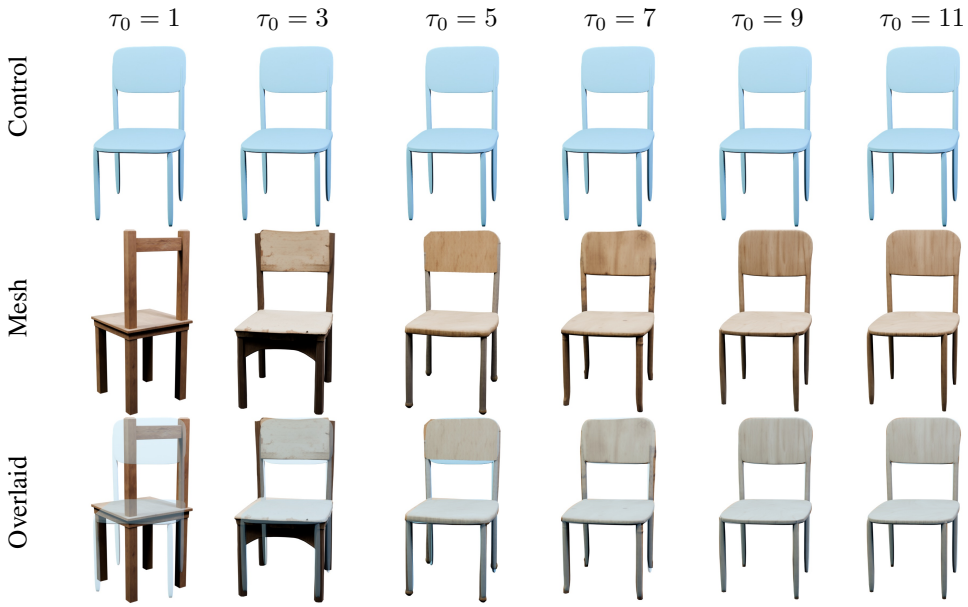


Figure 12: **Fine-grained alignment of SPACECONTROL with different τ_0.** In the first row we show the input control, in the second the generated asset and in the third, we overlay the two, to better visualize alignment. All the generations use the same spatial control and the same prompt *"A wooden chair."*.

Furthermore, in Fig. 13 we show a practical application when fine-grained spatial control can be particularly useful. With our method, a user can provide a sketch of the geometric primitives composing the scene and directly condition the generation on this input, without requiring any time-consuming post-processing to align the generated shapes.

Input spatial control | Output 3D assets

Figure 13: **SPACECONTROL for 3D scene generation.** We show how SPACECONTROL can be used to generate objects of full scenes starting from a coarse conditioning. On the left we show the superquadrics for the scene, where each object is represented with a different color. On the right we show the assets generated with SPACECONTROL using the geometric primitives from the right as spatial condition. Note that each object is generated independently, by scaling the superquadrics to unit cube and giving them as spatial control to SPACECONTROL. Generated objects are then automatically placed, by undoing the transformation.

B Interactive User Interface

In Fig. 14 we visualize our interactive user interface. Starting from scratch or from a template of superquadrics, users can freely edit superquadrics using their parameters, and add/delete them. Once given the conditioning, they can select a control strength (higher control strength means that the generated shape looks more like the primitives) and a text (and optionally image) conditioning. They can then toggle between the input primitives and meshes and proceed with new generations. We provide a demo of the user interface in the supplementary video.

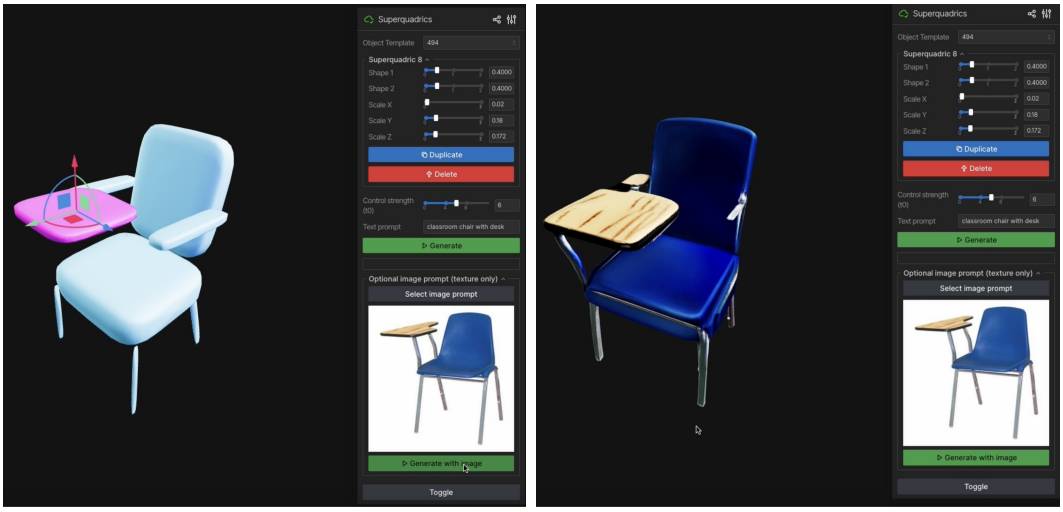


Figure 14: **Visualization of our interactive user interface.** Users can control the generated geometry by changing the shape of the geometric primitives and deciding the strength of the conditioning. Other than spatial control, users can use text and, optionally, images.

C User Study

In Fig. 15, we show the web interface of our user study. From left to right, we show the given control shape, and two competing methods. The participants then choose which generated object is more faithful to the input control shape, which model looks more realistic, and which one they like best.

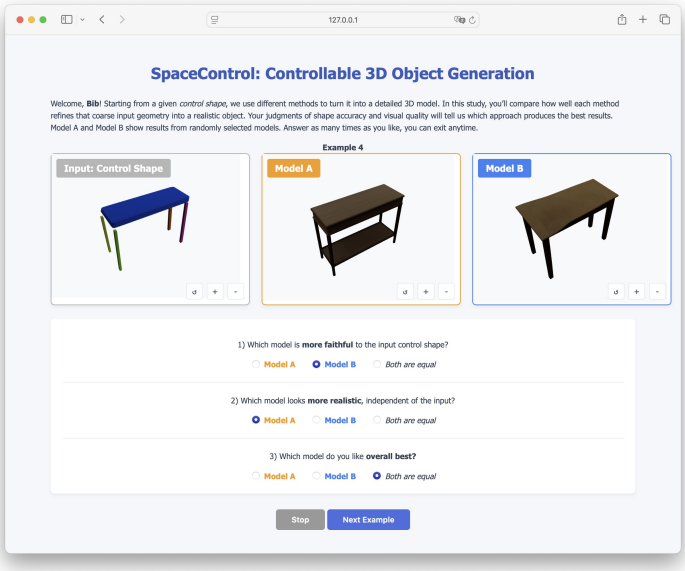


Figure 15: **User study interface.**

D SPICE-E-T

We obtain our training-based baseline SPICE-E-T by adding an additional conditioning layer to the flow transformer blocks in the structure generator of text-conditioned Trellis model (see Fig. 16) which perform cross attention on the shape conditioning. We encode the shape conditioning using the Trellis encoder $\mathcal{E}$, and we perform the Cross-Attention in that feature space. We initialize the original layers with the weights from the text-conditioned Trellis and the newly added ones randomly. We then train the modified *Structure Generator* for 120.000 iterations with a batch size of 4 on the ABO dataset (Collins et al., 2022), where the shape conditioning are obtained by running SuperDec (Fedele et al., 2025). During training, we use the same reconstruction loss of the original Trellis model.

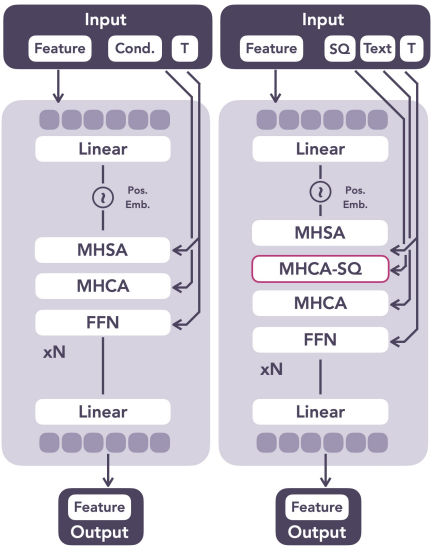


Figure 16: Comparison between the Flow Transformer from the original Trellis *(left)* and the one from SPICE-E-T *(right)*, adapted to enable spatial control via superquadrics.