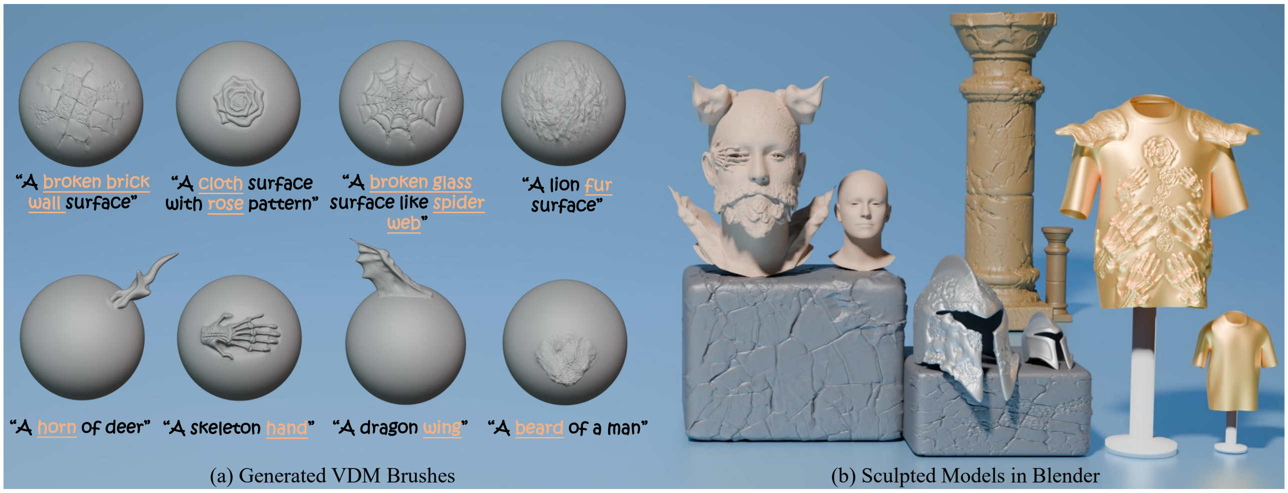
Text2VDM: Text to Vector Displacement Maps for Expressive and Interactive 3D Sculpting

Hengyu Meng[1] Duotun Wang[1] Zhijing Shao[1] Ligang Liu[2] Zeyu Wang[1,3]*

[1]The Hong Kong University of Science and Technology (Guangzhou)

[2]University of Science and Technology of China

[3]The Hong Kong University of Science and Technology


"A broken brick wall surface"
"A cloth surface with rose pattern"
"A broken glass surface like spider web"
"A lion fur surface"
"A horn of deer"
"A skeleton hand"
"A dragon wing"
"A beard of a man"

(a) Generated VDM Brushes

(b) Sculpted Models in Blender

Figure 1. **Example VDM brushes generated by Text2VDM and sculpted models in Blender.** Text2VDM can produce high-quality brushes for surface details (top row) and geometric structures (bottom row) from text input. Users can rapidly create an expressive model from a plain shape by directly applying these brushes in Blender. Yellow underlined text highlights semantics enhanced by our framework.

Abstract

Professional 3D asset creation often requires diverse sculpting brushes to add surface details and geometric structures. Despite recent progress in 3D generation, producing reusable sculpting brushes compatible with artists' workflows remains an open and challenging problem. These sculpting brushes are typically represented as vector displacement maps (VDMs), which existing models cannot easily generate compared to natural images. This paper presents Text2VDM, a novel framework for text-to-VDM brush generation through the deformation of a dense planar mesh guided by score distillation sampling (SDS). The original SDS loss is designed for generating full objects and struggles with generating desirable sub-object structures from scratch in brush generation. We refer to this issue as semantic coupling, which we address by introducing weighted blending of prompt tokens to SDS, resulting in a more accurate target distribution and semantic guidance. Experiments demonstrate that Text2VDM can generate diverse, high-quality VDM brushes for sculpting surface details and geometric structures. Our generated brushes can be seamlessly integrated into mainstream modeling software, enabling various applications such as mesh stylization and real-time interactive modeling.

1. Introduction

Sculpting brushes are essential tools in 3D asset creation, as artists often require a variety of brushes to create surface details and geometric structures. In modeling software, 3D sculpting brushes are typically defined as vector displacement maps (VDMs). A VDM is a 2D image where each pixel stores a 3D displacement vector. Through these vectors, VDM brushes can create complex surface details, such as cracks and wood grain, or generate geometric structures

*Corresponding author.

like ears and horns. This allows artists to apply the same geometric pattern iteratively while sculpting.

Despite significant advances in text-to-image (T2I) [36, 41] and text-to-3D generation [23, 30, 37, 49, 50], existing methods are unsuitable for creating VDM brushes. We summarize the challenges as follows: 1) Since VDMs are not natural images (Figure 2), it is difficult for existing T2I models to generate them directly. 2) From a 3D perspective, a VDM represents mesh deformation through per-vertex displacement vectors from a dense planar mesh. Mapping any generated mesh to a dense planar mesh to create a VDM is non-trivial. 3) Sculpting brushes often involve sub-object structures, whereas most 3D generation methods can only generate full objects. Enabling users to accurately control the generation of sculpting brushes through text prompts in a semantically focused manner remains challenging.

To address the challenges of brush generation, we propose Text2VDM, a novel optimization-based framework that generates diverse and controllable VDM brushes from text input. Our approach does not generate VDMs directly from a T2I model. Instead, we address VDM brush generation from a 3D perspective by applying score distillation with a pre-trained T2I model to guide mesh deformation. Our framework supports three ways to initialize a base mesh through a zero-valued, spike-pattern, or user-specified VDM for custom shape control. For mesh deformation, we formulate a Sobolev preconditioned optimization [35] to maintain mesh quality with intrinsic smoothness. We also provide optional region control using a mask of activated mesh deformation, helping users obtain the intended brush effects. The normal maps of the mesh are then rasterized by a differentiable renderer for brush optimization.

We observed that the standard score distillation sampling (SDS) [37] can lead to semantic coupling when supervising the generation of sub-object level structures due to the associated semantics caused by the noisy gradients from the full object. For example, a generated deer's horn should not be a full deer's head, or a generated beard should not include a nose. A straightforward solution is to use negative prompts [18, 59] to exclude undesired semantics, but our experiments show that this semantics suppression approach is ineffective in decoupling semantics and leads to an unstable optimization process. Instead, we propose to enhance the semantics of part-related words by applying weighted blending to the tokens in the prompt. This results in semantically focused text embedding, directing toward a more precise target distribution while reducing noisy gradients during optimization.

Our experiments demonstrate that Text2VDM produces high-quality and diverse VDM brushes that can be directly integrated into mainstream modeling software, such as Blender [5] and ZBrush [12]. Compared to existing methods that directly generate full 3D models, our approach addresses a different use case where brush-based user sculpting is desirable. This enables artists to interactively use a variety of brushes to sculpt diverse and expressive models from a plain shape.

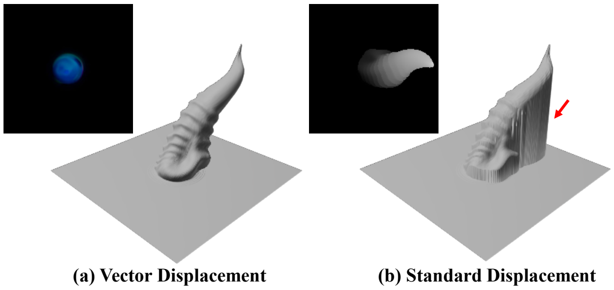


Figure 2. **Difference between vector and standard displacement.** The VDM enables full 3D vector displacement, while the height map only allows unidirectional standard displacement.

This paper makes the following contributions:

- We first introduce the task of text-to-VDM brush generation, which is challenging to tackle directly using current text-to-image and text-to-3D methods.
- We propose Text2VDM, a novel framework for text-to-VDM brush generation that is readily compatible with artists' workflow of 3D asset creation.
- We design a novel Semantic Enhancement SDS loss, which uses weighted blending to mitigate semantic coupling for sub-object structure generation.

2. Related Work

Text to Local 3D Generation and Editing. With recent advances in diffusion models [41] and differentiable 3D representations [1, 33, 35, 44, 46], many methods for text-guided full 3D model generation have emerged [8, 11, 23, 25, 31, 38, 42]. Since 3D content creation is an iterative process that often requires user interaction, more attention has been directed toward localized 3D generation and editing. For example, 3D Highlighter [9] and 3D Paintbrush [10] use text as input, leveraging pre-trained CLIP models [40] or diffusion models [37] to supervise the optimization of neural networks for segmenting the regions of a 3D model that match the text description. Based on the information from these segmented regions, further editing of texture and geometry can be applied to the 3D model. Furthermore, SKED [32] and SketchDream [27] introduce sketches as an additional modality to assist in localized editing. To enable more precise control, FocalDreamer [24], MagicClay [4], and Tip-Editor [60] allow users to specify the editing location directly within the 3D space. These works rely on optimization-based methods to edit specific objects, often resulting in non-reusable editing outcomes. Additionally, each edit requires a lengthy optimization process, making interactivity difficult to achieve.

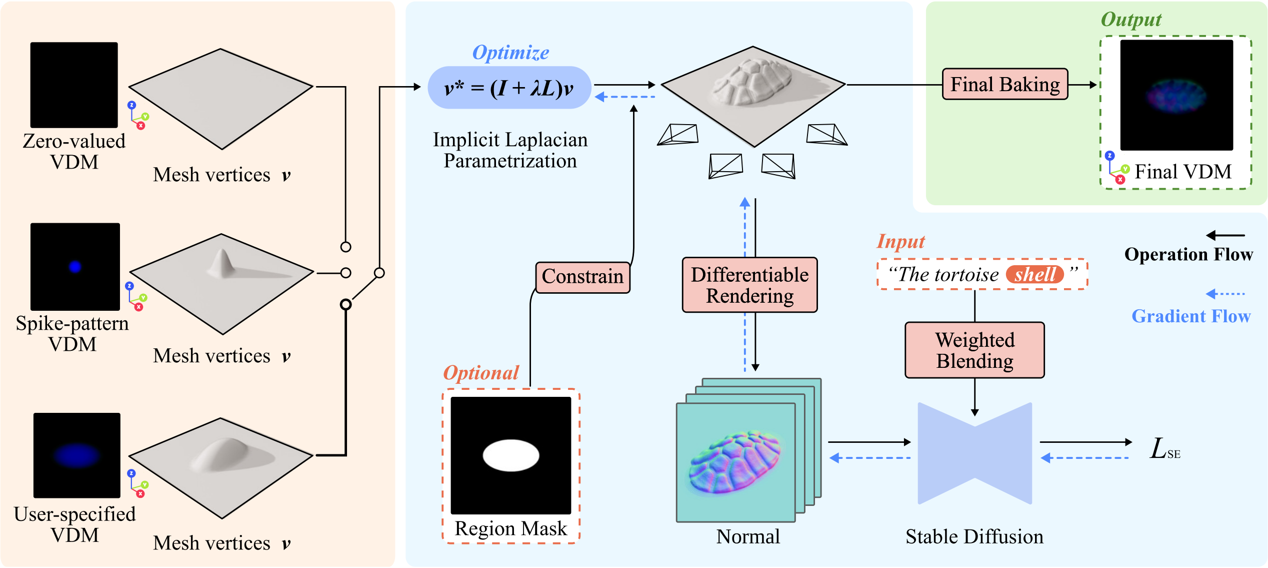


Figure 3. **Overview of Text2VDM.** Starting with a dense planar mesh constructed from a zero-valued VDM, users can initialize the mesh volume through a default spike-pattern VDM or a user-specified VDM. Given the text prompt and region mask, we propose the Semantic Enhancement SDS loss $\mathcal{L}_{\text{SE}}$ to guide mesh deformations through a mesh Laplacian L iteratively, achieving semantically focused generation of surface details or geometric structures. After generation, vertex displacements are baked into the final VDM.

Diffusion Priors for 3D Generation. Score distillation sampling (SDS) [37, 52] provides pixel-level guidance by seeking specific modes in a diffusion model, inspiring further research to improve optimization-based 3D generation [3, 26, 53, 54, 56]. Some studies focus on mitigating the "Janus" problem [2, 15], while others fine-tune diffusion models with multiview datasets to enhance 3D consistency [28, 45]. Recent research focuses on refining the design of SDS loss to achieve more precise guidance. For instance, Make-it-3D [50] introduces two-stage optimizations to improve textured appearance, while Fantasia3D [8] dynamically modifies the time-dependent weighting function within SDS computations. Additionally, several methods [18, 59] incorporate negative prompts as the conditional term to further refine the optimizations. Although diffusion priors have achieved promising results, their application in generating sub-object structures without global context as a reference is still challenging.

Appearance and Geometric Brush Synthesis. The concept of brushes is very common in the creative process of digital artists, serving as a reusable local decorative unit. Appearance brushes focus on color representation and drawing styles in 2D space. With the development of generative models [13, 41], many works have explored the synthesis of procedural material [21, 22] for 3D object texturing and appearance brushes for interactive painting [16, 47], realistic artworks generation [29, 34, 61], and applying stylization [17, 19]. Unlike appearance brushes, geometric brushes focus on modifying geometry by moving the vertices of a mesh in 3D space. VDM brushes, as an extension of standard geometric brushes, provide more complex geometric effects by utilizing VDMs. To the best of our knowledge, only a few techniques adopted the concepts of VDM for generation [43, 55] and geometric texture transfer [14]. Recently, concurrent work [57] explored using a single image as input and leveraged a diffusion model for generating multiview normal maps to guide VDM reconstruction. In comparison, our method is fundamentally different in tackling the semantic coupling problem arising from text guidance. This demonstrates that generating geometric brushes is a highly promising research direction.

3. Methodology

To generate VDM brushes compatible with mainstream modeling software, we begin by constructing a dense mesh from an initial VDM, as shown in Figure 3. We then apply score distillation with Stable Diffusion to guide high-quality mesh deformation formulated as a form of Sobolev preconditioned optimization [35]. To produce the intended sub-object level structure described in the text, we design a Semantic Enhancement SDS loss by applying weighted blending to the tokens in the prompt, effectively handling the issue of semantic coupling in SDS.

3.1. Brush Initialization

We provide three methods to initialize a base mesh for brush generation via a zero-valued VDM, a spike-pattern VDM, or a user-specified VDM. A VDM is represented as a 512 ×

512 three-channel image, in which each channel stores the displacement in the X, Y, or Z direction, respectively. We first construct a planar grid mesh by creating two triangles for every 2×2 pixels and then apply the displacement stored in the VDM to mesh vertices. The values in these three initial VDMs range from 0 to 1, in which 0 represents no displacement, and 1 corresponds to half of the mesh's edge length in the positive axis direction. Since users can apply sculpting brushes symmetrically, our initial VDM does not need to store any negative values.

Our three methods for brush initialization facilitate the generation of diverse sculpting brush styles. The zero-valued VDM results in a planar mesh, which is our default setup when no control is provided. The spike-pattern VDM is suitable for generating protruding geometric structures, as it can effectively adjust the Laplacian term in Equation (2) to steer the gradient direction for mesh deformation. For better control of the brush's volume and direction, we also provide an interface for users to create custom VDMs, so the user-specified brush initialization can effectively guide mesh deformation toward the target structure.

3.2. Brush Generation via Mesh Deformation

Given the vertices v on the initialized base mesh, our method aims to learn a mesh deformation to the target brush shape. The vertex positions $\hat{v}$ after mesh deformation can be expressed by:

$$\hat{v} = \arg\min_{v} \mathcal{L}_{\text{SE}}(\mathcal{D}_c(v), y), \tag{1}$$

where c represents the camera setup in a differentiable renderer $\mathcal{D}$ [20]. The loss function $\mathcal{L}_{\text{SE}}$ receives the rendered normal image $\mathcal{D}_c(v)$ and text input y to evaluate the semantic guidance, which is detailed in Section 3.3. To accompany the external forces from $\mathcal{L}_{\text{SE}}$ that drive the mesh deformation with intrinsic smoothness energies, we follow the framework of a Sobolev preconditioned gradient descent [35], where the base mesh is reparameterized by the mesh Laplacian L:

$$v^* = (I + \lambda L)v. \tag{2}$$

This preconditioning involves solving a sparse linear system at every iteration, modifying the gradient descent update for each mesh deformation step to:

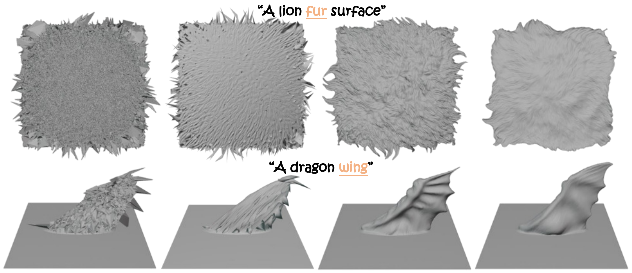
$$v \leftarrow v - \eta (I + \lambda L)^{-1} \frac{\partial \mathcal{L}_{\text{SE}}}{\partial v}, \tag{3}$$

where η is the learning rate, I is the identity matrix, and λ is a hyperparameter to control the extent of gradient diffusion over the entire domain. We set $\lambda = 15$ throughout our experiments to balance the global structure and fine details during mesh deformation.

Compared to directly adding a Laplacian regularization term to Equation (1), the preconditioning framework [35] is critical in achieving large mesh deformation while maintaining proper topology with reduced triangle flips (Figure 4). Several works [11, 51] adopt the strategy by Aigerman et al. [1], parameterizing deformation through Jacobian fields that capture the scaling and rotation of each triangle. Although this method effectively smooths vertex displacements, the local deformation represented in Jacobians accumulates, leading to global drifting for open-boundary meshes, making it challenging to bake the mesh as a brush.

Additionally, we provide an optional region mask to restrict mesh deformation to the user-defined region during optimization. It helps maintain zero values in unused VDM areas when generating geometric structures. For producing surface details, the region mask ensures that the brush effect meets user-customized requirements (Figure 9).

By framing the VDM generation task as mesh deformation through preconditioned optimization, our framework preserves the favorable initial mesh topology and ensures control throughout the procedure. The resulting mesh preserves the original structure while incorporating rich local deformations, making it well-suited for baking as a brush.



(a) λ=0 (b) λ=0 w/ Reg (c) λ=15 **(Ours)** (d) λ=30

Figure 4. **Effect of λ.** (a) Low-quality results w/o mesh smoothness, (b) low-quality results w/ direct Laplacian regularization, (c) our adopted preconditioning scheme with λ = 15, and (d) results with over-smoothness caused by a larger λ.

3.3. Semantic Enhancement Score Distillation

Current text-to-3D generation methods like DreamFusion [37] often optimize a 3D representation parameterized by θ so that rendered images $\mathbf{x} = g(\theta)$ resemble 2D samples produced by a pre-trained T2I diffusion model for a given text prompt y. g functions as a differentiable renderer. The T2I diffusion model ϕ predicts the sampled noise $\epsilon_\phi(\mathbf{x}_t; y, t)$ of a rendered image $\mathbf{x}_t$ at a noise level t for the text input y. To make rendered images follow the text-conditioned distribution in Stable Diffusion, the SDS loss updates θ by estimating the gradient:

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\phi, \mathbf{x}) = \mathbb{E}_{t,\epsilon,c} \left[\omega(t) \left(\epsilon_\phi(\mathbf{x}_t; y, t) - \epsilon \right) \frac{\partial \mathbf{x}}{\partial \theta} \right], \tag{4}$$

where $\omega(t)$ is a time-dependent weighting function.

However, the SDS loss cannot effectively supervise sub-object structure generation due to the issue of semantic coupling in full objects. For example, when using the SDS loss to generate a tortoise shell, it also usually generates the tortoise's tail and head, causing semantic coupling (Figure 7). We believe that the issue of semantic coupling in SDS stems from the training data of Stable Diffusion, in which most images contain full objects rather than separate parts. Therefore, the semantics of full objects often appear in the target distribution conditioned on text only describing sub-object structures.

A straightforward approach is using negative distributions via Classifier Score Distillation (CSD) [59] or Variational Score Distillation (VSD) [54] to suppress coupled semantics. CSD employs predefined negative prompts, resulting in more accurate negative distributions than those adaptively learned by VSD [59]:

$$\begin{aligned}\nabla_\theta \mathcal{L}_{\text{CSD}}(\phi,\mathbf{x}) = \mathbb{E}_{t,\epsilon,c}[(&\omega_{\text{pos}}\cdot\epsilon_\phi(\mathbf{x}_t;y,t)\\ &-\omega_{\text{neg}}\cdot\epsilon_\phi(\mathbf{x}_t;y_{\text{neg}},t))\frac{\partial \mathbf{x}}{\partial\theta}],\end{aligned} \tag{5}$$

where ω_{pos} and ω_{neg} denote different weights for positive and negative prompts. It requires two separate inferences with positive and negative prompts to obtain two distributions, which are then subtracted to suppress coupled semantics. However, our experiments show that CSD is ineffective in decoupling semantics because the negative prompt cannot accurately represent the undesirable coupled semantics in the positive prompt (Figure 7). This results in noisy guidance, making CSD less effective in decoupling semantics.

Unlike semantic suppression in CSD, we propose a semantic enhancement method to mitigate semantic coupling by enhancing the semantics of part-related words. This can lead to a more accurate and stable target distribution, as shown in Section 4.3. Our key design is to apply weighted blending to the tokens in the original prompt to obtain a semantically focused text embedding, which serves as stable guidance for the optimization process. We define the Semantic Enhancement SDS loss as:

$$\nabla_\theta \mathcal{L}_{\text{SE}}(\phi,\mathbf{x}) = \mathbb{E}_{t,\epsilon,c}\left[\omega(t)\left(\epsilon^*_\phi(\mathbf{x}_t;y,t)-\epsilon\right)\frac{\partial\mathbf{x}}{\partial\theta}\right], \tag{6}$$

where $\epsilon^*_\phi(\cdot)$ uses a text embedding weighted by Compel [48]. Specifically, we assign a weight s to each word in the prompt and compute the weighted embedding e_{w} for each word by blending the original word embedding e and the empty text embedding e_0: $e_{\text{w}} = e_0 + s\cdot(e-e_0)$. By concatenating the weighted embedding of each word in sequence, we obtain the final semantically focused text embedding. In our experiments, we found that using $s = 1.1^2$ for words that require semantic enhancement can achieve stable optimization and effectively alleviate the issue of semantic coupling.

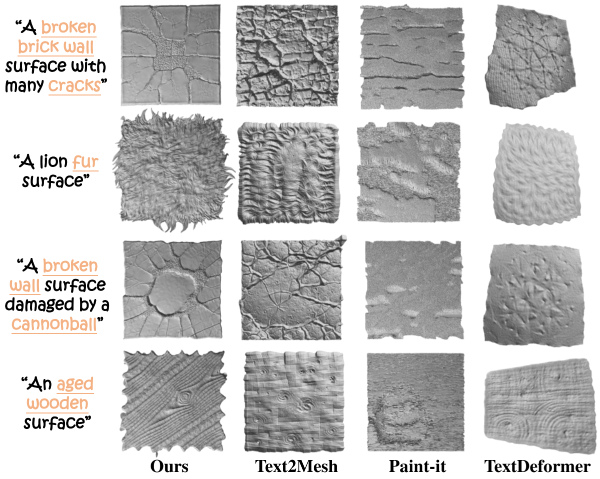

"A broken brick wall surface with many cracks"
"A lion fur surface"
"A broken wall surface damaged by a cannonball"
"An aged wooden surface"
Ours
Text2Mesh
Paint-it
TextDeformer

Figure 5. **Qualitative comparisons of generated brushes for surface details.** Our method captures geometry details guided by text, effectively preserving the surface structure and avoiding mesh distortion.

4. Experiments

We conducted experiments to evaluate the various capabilities of Text2VDM both quantitatively and qualitatively for text-to-VDM brush generation. We then present an ablation study that validates the significance of our key insight into Semantic Enhancement SDS, as well as the effect of the region mask and VDM initialization.

4.1. Qualitative Evaluation

To the best of our knowledge, Text2VDM is the first framework to generate VDM brushes from text. We adapted three existing methods for comparison and classified them into two categories. The first category includes Text2Mesh [31] and TextDeformer [11], which generate a brush mesh through text-guided mesh deformation on a planar mesh, following a process similar to ours. For the second category, we opt to directly generate VDM via Paint-it [58]. Notably, this method originally uses SDS to optimize a UNet for generating PBR textures. We reframed it to suit our VDM brush generation task, modifying it to generate VDM through SDS optimization of the UNet. In geometric structures generation experiment (Figure 6), all methods are compared fairly, with the same non-zero VDM initialization and mask applied to each prompt (see Appendix B.2). For surface details (Figure 5), all methods start with a zero-valued VDM and no masks to ensure a fair comparison.

Compared to other methods, Text2VDM can generate better-quality VDM brushes. Text2Mesh applies displacement to each vertex along normal directions, resulting in limited mesh deformation. TextDeformer indicates the accumulation of local deformations in the Jacobians, which results in global mesh drift, making it challenging to bake

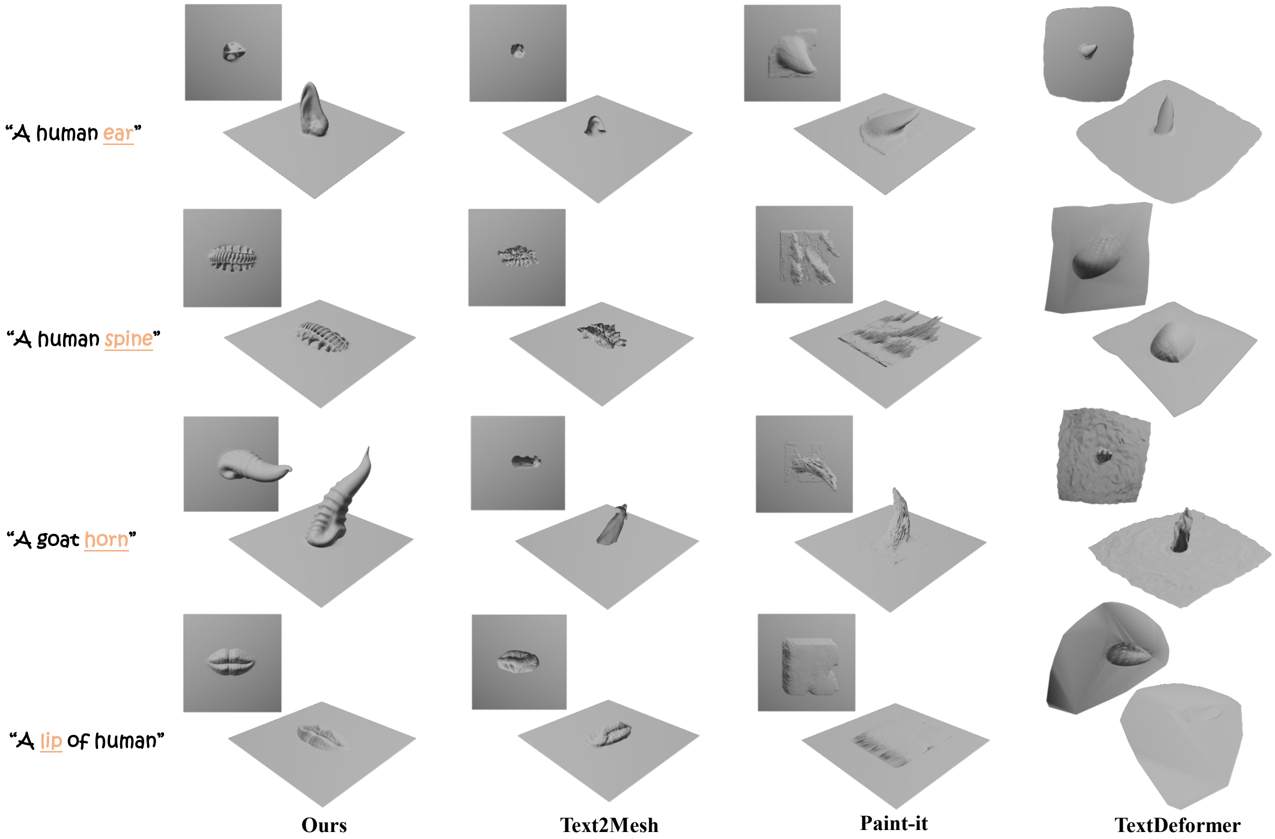


Figure 6. **Qualitative comparisons of generated brushes for geometric structures.** Our method accurately presents key geometric features described by text, facilitating downstream applications in modeling software.

these meshes into VDM. Reframed Paint-it VDM generation is equivalent to optimizing the three-axis displacement of each vertex on the mesh with SDS. Although the UNet reduces noise from the SDS [58], smooth regularization is still required to ensure mesh quality, which makes achieving high-quality mesh generation quite challenging.

4.2. Quantitative Evaluation

We quantitatively evaluated our framework regarding generation consistency with text input and mesh quality. We used 40 distinctive text prompts for VDM generation.

Generation Consistency with Text. We initially assessed the relevance of the generated results to the text descriptions [40]. 12 different views were rendered for average scores respectively, as presented in Table 1. Our approach achieves the highest scores compared to baseline methods.

Mesh Quality. We evaluated mesh quality by examining self-intersection. Paint-it and Text2Mesh, which utilize direct vertex displacement, often converge to a local minimum and disregard the mesh triangulation. While TextDeformer exhibits the lowest self-intersection, its tendency to produce over-smoothed results frequently results in losing object features described in text prompts.

Table 1. Quantitative evaluation of state-of-the-art methods. The geometry CLIP score is calculated on shaded images with uniform albedo colors [39], and self-intersection is quantified as the ratio of self-intersected mesh faces to the total number of faces.

	Geometry CLIP Score ↑	Mesh Self-Intersection ↓
Paint-it	0.2375	19.42%
Text2Mesh	0.2497	7.18%
TextDeformer	0.2477	**0.04%**
Ours	**0.2556**	0.77%

Table 2. User evaluation of generated VDMs.

User Preference ↑	Geometry Quality	Consistency with Text
Paint-it	3.1%	1.7%
Text2Mesh	18.3%	27.3%
TextDeformer	3.3%	3.4%
Ours	**75.3%**	**67.6%**

User Study. We further conducted a user study to evaluate the effectiveness and expressiveness of our method. A Google Form was utilized to assess 1) geometry quality and 2) consistency with text. We recruited 32 participants, of whom 14 are graduate students majoring in media arts, and 18 are company employees specializing in AI content gen-

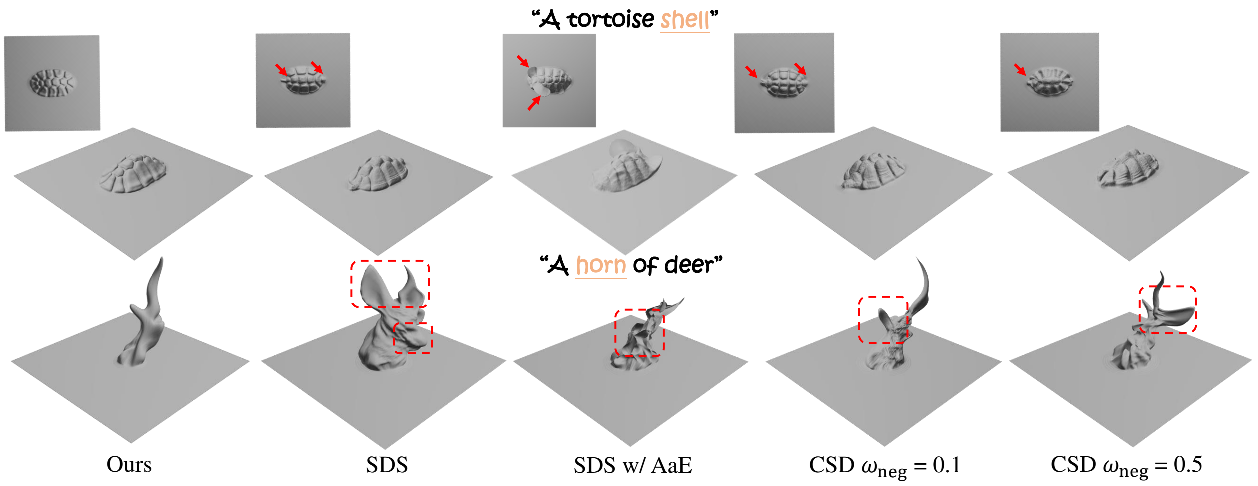


Figure 7. **Effect of Semantic Enhancement SDS.** Our method effectively mitigates semantic coupling issues in SDS, such as generating the tortoise's tail and head or the deer's ear and mouth, by providing more focused semantic guidance. In contrast, the semantic suppression method used in CSD and the other semantic enhancement approach proposed by Attend-and-Excite both lead to an unstable optimization.

eration. The participants were instructed to choose the preferred renderings of VDM from different methods in randomized order, as shown in Table 2. The results show participants preferred our method by a significant margin.

4.3. Ablation Study

Effect of Semantic Enhancement SDS. Figure 7 compares the results generated by the original SDS [37], our Semantic Enhancement SDS with Compel, SDS with another semantic enhancement method in Attend-and-Excite (AaE) [7], and CSD [59] with two different annealed weights for the negative prompts: "tortoise tail, tortoise head" and "deer's ear, deer's mouth." We also qualitatively compare these methods by visualizing their performance on semantic decoupling using the same Stable Diffusion for 2D image generation (Figure 8). Compared to semantic suppression using negative prompts and semantic enhancement by AaE, our design of introducing a semantically focused text embedding via Compel to SDS is most effective in resolving the issue of semantic coupling.

Key Insight. As discussed in Section 3.3, SDS can result in semantic coupling when generating sub-object structures, leading to artifacts like the tortoise's tail and head or the deer's ear and deer's mouth. We observed that the semantics represented by negative prompts are also coupled. Therefore, meaningless semantics can emerge when applying semantic suppression in SDS, which leads to unstable optimization. Increasing the weight of negative prompts further reduces the overall quality of generated results. AaE enhances the cross-attention map of specific tokens by continuously updating the latent code. However, AaE is not suitable for the SDS framework because the latent code is affected by different camera poses and random noise at each iteration. This temporal randomness undermines the continuity of the latent code, resulting in unstable optimization and subpar results. In contrast, our method uses Compel to enhance semantics and achieves more effective semantic decoupling. Moreover, the semantically focused text embedding produced by Compel is independent of temporal variation. These properties help mitigate semantic coupling during SDS optimization, leading to high-quality sub-object geometric structures. Meanwhile, our method can also enhance the semantics in the text prompt for surface detail generation (see Appendix A.2).

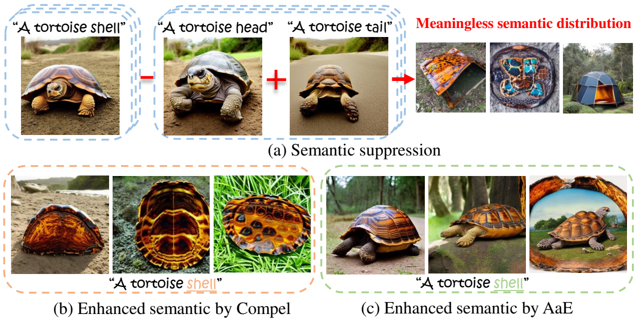


Figure 8. **Visualization of semantic decoupling performance.** The images generated by the T2I model qualitatively visualize sub-object semantics. (a) Coupled semantics in positive and negative prompts lead to meaningless distributions when subtracted. (b) Enhanced semantics by Compel achieves stable decoupling. (c) AaE can only ensure that the enhanced semantics are preserved, but cannot decouple negative ones effectively.

Effect of Region Mask. Figure 9 demonstrates two region masks and their control over surface detail generation given a text prompt. Without a region mask, the results can still

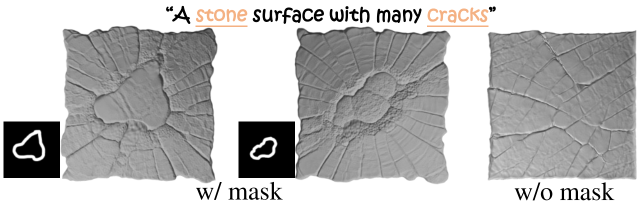


Figure 9. **Effect of region mask.** Region masks can effectively control the pattern of surface details based on text input.

match the text prompt but lack a specific pattern. By using a region mask and maintaining an activation ratio of 1/2 throughout the total iterations as a warm-up stage, we achieve a reasonable tradeoff between mask-result alignment and generation diversity (see Appendix A.3).

Effect of VDM Initialization. Our method demonstrates that user-specified VDMs can effectively control the volume and direction of generated geometric structures. Figure 10 shows that the results are high-quality and match the text descriptions well, such as the elf ear and pauldron. As this initializes the Laplacian term and steers the gradient flow in geometric structure generation, users can easily specify an initial VDM or choose a VDM template provided in our framework to generate expressive results.

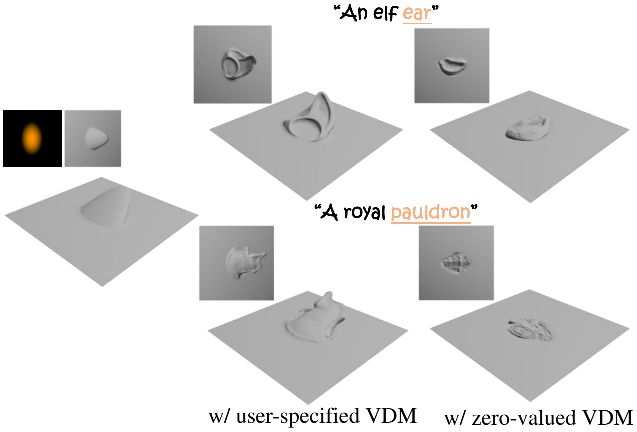


Figure 10. **Effect of VDM initialization.** User-specified VDMs can help achieve the intended final effect of geometric structures by initializing the brush's volume and direction.

4.4. Applications

Once various VDMs are generated, users can use these brushes to meet diverse creative needs in modeling software. For example, they can apply VDM brushes for mesh stylization and a real-time iterative modeling process.

Local-to-Global Mesh Stylization. Although mesh stylization is a complex task even for professional artists, combining different surface details allows users to achieve stylization quickly (see Appendix C.3).

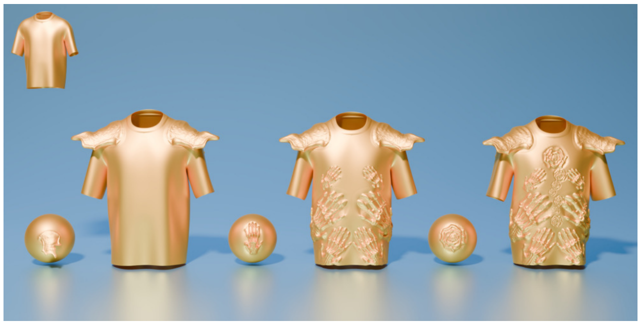


Figure 11. **Coarse to fine interactive modeling.** By combining geometric structures brushes and surface details brushes for iterative sculpting in modeling software, users can rapidly create an expressive model from a plain shape (top left).

Coarse-to-Fine Interactive Modeling. Unlike previous methods [4, 60] that require a lengthy optimization process for each edit and result in non-reusable outcomes, our generated VDM brushes can be directly used in modeling software. This enables users to apply the generated brushes easily and interactively (Figure 11).

5. Conclusion

We have presented Text2VDM, a novel framework for VDM brush generation from text. A VDM is a non-natural 2D image where each pixel stores a 3D displacement vector, making it challenging for existing T2I models to generate. Therefore, we treat VDM generation as diffusion-guided mesh deformation formulated as a form of Sobolev preconditioned optimization. To mitigate semantic coupling issues in SDS, we propose using weighted blending for prompt tokens, achieving high-quality brush generation. Moreover, we introduce two control methods, i.e., region and shape control, to meet customized requirements. The generated VDMs are directly compatible with mainstream modeling software, enabling various applications such as mesh stylization and interactive modeling.

Limitations and Future Work. While our framework can generate high-quality VDM brushes, they may encounter multiview inconsistencies, a common issue introduced by SDS. View-consistent diffusion models like MV2MV [6] may be helpful to further handle this. Additionally, errors of the T2I models may guide wrong 3D generation, such as the wrong number of fingers in the generated skeleton hand. VDMs demonstrate that complex 3D models can be efficiently created using diverse reusable sculpting brushes. Future exploration of 3D generation through the assembly of modular components with similar design principles holds promising research value.

Acknowledgments

This work was partially supported by the Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things #2023B1212010007.

References

[1] Noam Aigerman, Kunal Gupta, Vladimir G. Kim, Siddhartha Chaudhuri, Jun Saito, and Thibault Groueix. Neural Jacobian Fields: Learning Intrinsic Mappings of Arbitrary Meshes. *ACM Trans. Graph.*, 41(4), 2022. 2, 4

[2] Thiemo Alldieck, Nikos Kolotouros, and Cristian Sminchisescu. Score Distillation Sampling with Learned Manifold Corrective. *European Conference on Computer Vision*, 2024. 3

[3] Mohammadreza Armandpour, Ali Sadeghian, Huangjie Zheng, Amir Sadeghian, and Mingyuan Zhou. Re-imagine the Negative Prompt Algorithm: Transform 2D Diffusion into 3D, alleviate Janus problem and Beyond. *arXiv preprint 2304.04968*, 2023. 3

[4] Amir Barda, Vladimir G. Kim, Noam Aigerman, Amit H. Bermano, and Thibault Groueix. MagicClay: Sculpting Meshes with Generative Neural Fields. *SIGGRAPH Asia (Conference track)*, 2024. 2, 8

[5] Blender Foundation. Blender. https://www.blender.org, 2025. Accessed Mar 5, 2025. 2

[6] Youcheng Cai, Runshi Li, and Ligang Liu. MV2MV: Multi-View Image Translation via View-Consistent Diffusion Models. *ACM Trans. Graph.*, 2024. 8

[7] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-Excite: Attention-Based Semantic Guidance for Text-to-Image Diffusion Models. 42 (4), 2023. 7

[8] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3D: Disentangling Geometry and Appearance for High-quality Text-to-3D Content Creation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22246–22256, 2023. 2, 3

[9] Dale Decatur, Itai Lang, and Rana Hanocka. 3D Highlighter: Localizing Regions on 3D Shapes via Text Descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20930–20939, 2023. 2

[10] Dale Decatur, Itai Lang, Kfir Aberman, and Rana Hanocka. 3D Paintbrush: Local Stylization of 3D Shapes with Cascaded Score Distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4473–4483, 2024. 2

[11] William Gao, Noam Aigerman, Thibault Groueix, Vova Kim, and Rana Hanocka. TextDeformer: Geometry Manipulation Using Text Guidance. In *ACM SIGGRAPH 2023 Conference Proceedings*, New York, NY, USA, 2023. Association for Computing Machinery. 2, 4, 5

[12] Maxon Computer GMBH. ZBrush. https://www.maxon.net/zbrush, 1999. Accessed Oct 18, 2024. 2

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 27, 2014. 3

[14] Amir Hertz, Rana Hanocka, Raja Giryes, and Daniel Cohen-Or. Deep Geometric Texture Synthesis. *ACM Trans. Graph.*, 39(4), 2020. 3

[15] Susung Hong, Donghoon Ahn, and Seungryong Kim. Debiasing Scores and Prompts of 2D Diffusion for View-consistent Text-to-3D Generation. In *Neural Information Processing Systems*, 2023. 3

[16] Anita Hu, Nishkrit Desai, Hassan Abu Alhaija, Seung Wook Kim, and Maria Shugrina. Diffusion Texture Painting. New York, NY, USA, 2024. Association for Computing Machinery. 3

[17] Teng Hu, Ran Yi, Haokun Zhu, Liang Liu, Jinlong Peng, Yabiao Wang, Chengjie Wang, and Lizhuang Ma. Stroke-based Neural Painting and Stylization with Dynamically Predicted Painting Region. New York, NY, USA, 2023. Association for Computing Machinery. 3

[18] Oren Katzir, Or Patashnik, Daniel Cohen-Or, and Dani Lischinski. Noise-free Score Distillation. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 3

[19] Dmytro Kotovenko, Matthias Wright, Arthur Heimbrecht, and Björn Ommer. Rethinking Style Transfer: From Pixels to Parameterized Brushstrokes. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12191–12200, 2021. 3

[20] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular Primitives for High-Performance Differentiable Rendering. *ACM Transactions on Graphics*, 39(6), 2020. 4

[21] Beichen Li, Liang Shi, and Wojciech Matusik. End-to-end procedural material capture with proxy-free mixed-integer optimization. *ACM Transactions on Graphics (TOG)*, 42(4): 1–15, 2023. 3

[22] Beichen Li, Yiwei Hu, Paul Guerrero, Miloš Hašan, Liang Shi, Valentin Deschaintre, and Wojciech Matusik. Procedural material generation with reinforcement learning. *ACM Transactions on Graphics (TOG)*, 43(6):1–14, 2024. 3

[23] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3D: Fast Text-to-3D with Sparse-View Generation and Large Reconstruction Model. *arXiv preprint 2311.06214*, 2023. 2

[24] Yuhan Li, Yishun Dou, Yue Shi, Yu Lei, Xuanhong Chen, Yi Zhang, Peng Zhou, and Bingbing Ni. FocalDreamer: Text-Driven 3D Editing via Focal-Fusion Assembly. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38:3279–3287, 2024. 2

[25] Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. LucidDreamer: Towards High-Fidelity Text-to-3D Generation via Interval Score Matching. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6517–6526, 2023. 2

[26] 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 *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 300–309, 2023. 3

[27] Feng-Lin Liu, Hongbo Fu, Yu-Kun Lai, and Lin Gao. SketchDream: Sketch-based Text-to-3D Generation and Editing. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2024)*, 43(4), 2024. 2

[28] 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. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9264–9275, 2023. 3

[29] Songhua Liu, Tianwei Lin, Dongliang He, Fu Li, Ruifeng Deng, Xin Li, Errui Ding, and Hao Wang. Paint Transformer: Feed Forward Neural Painting with Stroke Prediction. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6578–6587, 2021. 3

[30] Xiaoxiao Long, Yuanchen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, and Wenping Wang. Wonder3D: Single Image to 3D Using Cross-Domain Diffusion. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9970–9980, 2023. 2

[31] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2Mesh: Text-Driven Neural Stylization for Meshes. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13482–13492, 2022. 2, 5

[32] Aryan Mikaeili, Or Perel, Mehdi Safaee, Daniel Cohen-Or, and Ali Mahdavi-Amiri. Sked: Sketch-guided text-based 3d editing. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2

[33] 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*, pages 405–421, 2020. 2

[34] Haoran Mo, Edgar Simo-Serra, Chengying Gao, Changqing Zou, and Ruomei Wang. General virtual sketching framework for vector line art. *ACM Transactions on Graphics (TOG)*, 40:1 – 14, 2021. 3

[35] Baptiste Nicolet, Alec Jacobson, and Wenzel Jakob. Large Steps in Inverse Rendering of Geometry. *ACM Trans. Graph.*, 40(6), 2021. 2, 3, 4

[36] Openai. DALLE. `https://openai.com/index/dall-e-3/`, 2024. Accessed Oct 22, 2024. 2

[37] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D Diffusion. In *The Eleventh International Conference on Learning Representations*, 2022. 2, 3, 4, 7

[38] Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi Zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. RichDreamer: A Generalizable Normal-Depth Diffusion Model for Detail Richness in Text-to-3D. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9914–9925, 2023. 2

[39] Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi Zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. RichDreamer: A Generalizable Normal-Depth Diffusion Model for Detail Richness in Text-to-3D. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9914–9925, 2024. 6

[40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning*, 2021. 2, 6

[41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis With Latent Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 2, 3

[42] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18603–18613, 2022. 2

[43] Artem Sevastopolsky, Philip-William Grassal, Simon Giebenhain, ShahRukh Athar, Luisa Verdoliva, and Matthias Niessner. Headcraft: Modeling high-detail shape variations for animated 3dmms, 2023. 3

[44] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep Marching Tetrahedra: a Hybrid Representation for High-resolution 3D Shape Synthesis. *Advances in Neural Information Processing Systems*, 34:6087–6101, 2021. 2

[45] Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. MVDream: Multi-view Diffusion for 3D Generation. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[46] J. Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3D Neural Field Generation Using Triplane Diffusion. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20875–20886, 2023. 2

[47] Maria Shugrina, Chin-Ying Li, and Sanja Fidler. Neural Brushstroke Engine: Learning a Latent Style Space of Interactive Drawing Tools. *ACM Trans. Graph.*, 41(6), 2022. 3

[48] Damian Stewart. Compel: A Text Prompt Weighting and Blending Library. `https://github.com/damian0815/compel`, 2023. Accessed Oct 20, 2024. 5

[49] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. DreamGaussian: Generative Gaussian Splatting for Efficient 3D Content Creation, 2023. 2

[50] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22819–22829, 2023. 2, 3

[51] Duotun Wang, Hengyu Meng, Zeyu Cai, Zhijing Shao, Qianxi Liu, Lin Wang, Mingming Fan, Xiaohang Zhan, and

Zeyu Wang. HeadEvolver: Text to Head Avatars via Expressive and Attribute-Preserving Mesh Deformation. *International Conference on 3D Vision*, 2025. 4

[52] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score Jacobian Chaining: Lifting Pretrained 2D Diffusion Models for 3D Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12619–12629, 2023. 3

[53] Peihao Wang, Dejia Xu, Zhiwen Fan, Dilin Wang, Sreyas Mohan, Forrest N. Iandola, Rakesh Ranjan, Yilei Li, Qiang Liu, Zhangyang Wang, and Vikas Chandra. Taming Mode Collapse in Score Distillation for Text-to-3D Generation. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

[54] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. ProlificDreamer: High-fidelity and Diverse Text-to-3D Generation with Variational Score Distillation. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2024. Curran Associates Inc. 3, 5

[55] Xingguang Yan, Han-Hung Lee, Ziyu Wan, and Angel X. Chang. An Object is Worth 64x64 Pixels: Generating 3D Object via Image Diffusion, 2024. 3

[56] Xiaofeng Yang, Yiwen Chen, Cheng Chen, Chi Zhang, Yi Xu, Xulei Yang, Fayao Liu, and Guosheng Lin. Learn to Optimize Denoising Scores for 3D Generation: A Unified and Improved Diffusion Prior on NeRF and 3D Gaussian Splatting. *arXiv preprint 2312.04820*, 2023. 3

[57] Yuezhi Yang, Qimin Chen, Vladimir Kim, Siddhartha Chaudhuri, Qixing Huang, and Zhiqin Chen. GenVDM: Generating Vector Displacement Maps From a Single Image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3

[58] Kim Youwang, Tae-Hyun Oh, and Gerard Pons-Moll. Paint-it: Text-to-Texture Synthesis via Deep Convolutional Texture Map Optimization and Physically-Based Rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 5, 6

[59] Xin Yu, Yuanchen Guo, Yangguang Li, Ding Liang, Song-Hai Zhang, and Xiaojuan Qi. Text-to-3D with Classifier Score Distillation. *arXiv preprint 2310.19415*, 2023. 2, 3, 5, 7

[60] Jingyu Zhuang, Di Kang, Yan-Pei Cao, Guanbin Li, Liang Lin, and Ying Shan. TIP-Editor: An Accurate 3D Editor Following Both Text-Prompts and Image-Prompts. *ACM Trans. Graph.*, 43(4), 2024. 2, 8

[61] Zhengxia Zou, Tianyang Shi, Shuang Qiu, Yi Yuan, and Zhenwei Shi. Stylized Neural Painting. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15684–15693, 2021. 3