SPLATFONT3D: STRUCTURE-AWARE TEXT-TO-3D ARTISTIC FONT GENERATION WITH PART-LEVEL STYLE CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Artistic font generation (AFG) can assist human designers in creating innovative artistic fonts. However, most previous studies primarily focus on 2D artistic fonts in flat design, leaving personalized 3D-AFG largely underexplored. 3D-AFG not only enables applications in immersive 3D environments such as video games and animations, but also may enhance 2D-AFG by rendering 2D fonts of novel views. Moreover, unlike general 3D objects, 3D fonts exhibit precise semantics with strong structural constraints and also demand fine-grained part-level style control. To address these challenges, we propose SplatFont3D, a novel structure-aware text-to-3D AFG framework with 3D Gaussian splatting, which enables the creation of 3D artistic fonts from diverse style text prompts with precise part-level style control. Specifically, we first introduce a Glyph2Cloud module, which progressively enhances both the shapes and styles of 2D glyphs (or components) and produces their corresponding 3D point clouds for Gaussian initialization. The initialized 3D Gaussians are further optimized through interaction with a pretrained 2D diffusion model using score distillation sampling. To enable part-level control, we present a dynamic component assignment strategy that exploits the geometric priors of 3D Gaussians to partition components, while alleviating drift-induced entanglement during 3D Gaussian optimization. Our SplatFont3D provides more explicit and effective part-level style control than NeRF, attaining faster rendering efficiency. Experiments show that our SplatFont3D outperforms existing 3D models for 3D-AFG in style–text consistency, visual quality, and rendering efficiency.

1 INTRODUCTION

Artistic fonts are widely used in movie posters, brand icons, video games, and many other areas in our daily lives. Different from standard printed fonts in books and computers, artistic fonts attain significant diversity in glyph shapes and font effects. It generally requires expert human designers to create personalized artistic fonts depending on specific scenarios or contexts, which is highly demanding in both time and financial cost. Therefore, there is a pressing need for methods of Artistic Font Generation (AFG), which can teach machines to automatically generate artistic fonts. Such innovative techniques are supposed to assist human designers in creating customized 3D artistic fonts. With recent advances in GANs and diffusion models (Goodfellow et al., 2020; Ho et al., 2020), AFG has achieved remarkable success (Hayashi et al., 2019; Wang et al., 2023a; Li et al., 2023b; Miao et al., 2024; Mu et al., 2024; Ren et al., 2025).

Although previous studies are capable of generating novel 2D collections by combining various existing glyphs and textures, they are primarily limited to 2D artistic fonts in flat design, leaving 3D font synthesis largely underexplored. Compared with 2D-AFG, 3D-AFG offers broader application prospects and greater practical values. For example, most 2D-AFG methods are confined to creating 2D planar images from a pre-defined viewpoint but are not capable of generating novel views, limiting their flexibility and practicability. Instead, 3D artistic fonts can explicitly represent the spatial structures of fonts and can feasibly render 2D fonts of arbitrary views, thereby positioning 2D-AFG as a special case of its 3D counterpart. Moreover, 3D-AFG enables applications in immersive 3D environments such as 3D animations, video games, and virtual reality. Therefore, 3D-AFG exhibits significantly better application potential than 2D approaches, which is worth further investigation.

Nevertheless, 3D-AFG poses unique challenges beyond the general 3D object synthesis, essentially making existing text-to-3D models inapplicable. Specifically,

1. **Semantic & Style Constraints.** Different from general objects, character fonts encode rich semantic information, and their shapes are strictly constrained to preserve semantic correctness. However, existing pre-trained text-to-3D models (Poole et al., 2023; Lin et al., 2023; Wang et al., 2023b; Yi et al., 2024a; Chen et al., 2024b; Huang et al., 2024; Liu et al., 2023b;a) or 2D diffusion models (Ho et al., 2020; Song et al., 2020; Rombach et al., 2022; Nichol et al., 2021) are primarily exposed to general objects, making them struggle with font recognition and understanding. This makes synthesizing 3D artistic fonts particularly challenging, especially in cases that require both preserving correct semantics under shape constraints and incorporating accurate stylistic attributes at precise layout positions.
2. **Part-level Style Control.** A more practical 3D-AFG model should go beyond the global stylization and further achieve structure-aware synthesis with part-level control. However, part-level modification is considerably difficult for existing 3D models, highlighting their limitations for structure-aware 3D-AFG. For example, NeRF (Mildenhall et al., 2021) represents objects implicitly using a neural field that essentially lacks natural decomposition, thus making the part modifications of 3D objects difficult. Moreover, 3DGS (Kerbl et al., 2023) represents objects with 3D Gaussian points, but it carries no precise semantics for reliable component partitioning.
3. **Expensive Acquisition Cost.** Unlike 2D images, 3D fonts are considerably scarce and not feasibly obtainable from publicly available sources (e.g., the Internet). Furthermore, the creation and collection of 3D artistic fonts present significant challenges, as they demand that designers possess formal expertise in artistic font design and mastery of 3D modeling software (e.g., Maya and 3ds Max). This substantially increases the time and financial cost as well as the complexity of dataset creation. The scarcity of 3D font datasets eventually makes it impractical to obtain a generalized, large-scale 3D-AFG model through the conventional supervised training.

So far, structure-aware 3D artistic font generation remains largely unexplored, and existing text-to-3D approaches (Poole et al., 2023; Lin et al., 2023; Wang et al., 2023b; Chen et al., 2023; Yi et al., 2024a; Chen et al., 2024b; Metzer et al., 2023; Huang et al., 2024; Liu et al., 2023b;a) still fail to address these challenges effectively. For example, a prevalent text-to-3D approach is to directly train 3D models on large-scale data collections, which, however, often struggles to effectively generalize into open-set domains. Consequently, the scarcity of 3D font data makes it infeasible to build a general-purpose, highly generalizable 3D-AFG model.

An alternative approach is to leverage large pre-trained 2D diffusion models for 3D generation (Poole et al., 2023; Lin et al., 2023; Wang et al., 2023b; Chen et al., 2023; Metzer et al., 2023; Shi et al., 2023; Yi et al., 2024a; Chen et al., 2024b), thereby avoiding the need for extensive data collection; however, significant challenges remain for 3D-AFG. Specifically, **(1)** a prior attempt at 3D-AFG, DreamFont3D (Li et al., 2024), leveraged a pre-trained 2D diffusion model to refine NeRF-based 3D volumes. However, due to their implicit representation, NeRF-based approaches struggle to achieve precise part-level control, as they lack structural decomposition of 3D fonts, and their rendering process remains highly time-consuming and computationally expensive. **(2)** Although 3DGS enables faster rendering than NeRF, its application to 3D-AFG remains challenging. This is because 3DGS requires well-initialized point clouds for high-quality generation, and fine-grained part-level control further depends on precise semantics for component portioning during optimization, which makes it difficult to achieve structure-aware 3D-AFG with existing 3D models.

To address those challenges, this paper proposes SplatFont3D, a novel structure-aware text-to-3D artistic font generation model with precise part-level style control, which leverages the geometric advantages of 3DGS and the strong prior knowledge of pre-trained 2D diffusion models. Specifically, **(1)** We first introduce a Glyph2Cloud module to progressively refine the geometry shapes of 2D glyphs (or components) while maintaining their semantics and further construct well-initialized 3D point clouds for Gaussian initialization. **(2)** The initialized 3D Gaussians further leverage the priors of 2D diffusion models and are accumulatively optimized via Score Distillation Sampling (SDS), which projects the differentiable 3D representation from various viewpoints and makes the projected 2D images match the text conditions. Such a strategy eliminates the need for acquiring real 3D artistic font data, effectively addressing the challenges posed by data scarcity. **(3)** To achieve part-level style control, we exploit the geometric priors of 3D Gaussians to partition components for individual rendering. However, the dynamic optimization of 3DGS often causes Gaussian points to drift, and

thus, points from different components may overlap and interfere with each other, ultimately degrading the generation quality. Hence, we further integrate a dynamic component assignment strategy to address this drift-induced component entanglement issue. This eventually enables more explicit and effective part-level style control than NeRF with faster rendering speeds. Experiments empirically demonstrate that our SplatFont3D can render 3D artistic fonts more effectively and efficiently than NeRF and existing text-to-3D models. Our contributions are summarized as follows:

- We propose SplatFont3D, a novel structure-aware text-to-3D artistic font generation model with precise part-level style control, a problem that has remained largely unexplored.
- We introduce Glyph2Cloud, a module that progressively refines the geometric shapes of 2D glyphs while maintaining original semantics, and consequently constructs well-initialized 3D point clouds for Gaussian initialization. This strategy enables the effective combination of 3DGS and 2D diffusion priors for 3D-AFG and further helps eliminate the need for acquiring real 3D font data, making the overall approach feasible.
- To enable precise part-level style control, we present dynamic component assignment that exploits the geometric priors of 3D Gaussians to partition components, while alleviating drift-induced entanglement during Gaussian optimization. Our explicit part-level control is more effective than the implicit one of NeRF, while attaining higher rendering efficiency.
- Extensive experiments demonstrate the superiority of our SplatFont3D over existing tex-to-3D models for 3D-AFG in style–text consistency, visual quality, and rendering efficiency.

2 Related Work

2.1 Artistic Font Generation

Artistic font generation (Gao et al., 2019; Li et al., 2022; Wang et al., 2023a) has emerged as a vibrant research area. Early studies approached AFG via conditional GANs (Goodfellow et al., 2020), such as zi2zi (Tian, 2017) and GlyphGAN (Hayashi et al., 2019), which enabled style-consistent transfer across character sets. Subsequent research explored component-aware and few-shot paradigms for capturing better intra-character structures. Chen et al. (2024a) explicitly modeled ideographic composition for characters, and Li et al. (2023b); Park et al. (2021) employed the attention and global–local disentanglement to synthesize characters from only a few exemplars. With the advent of large-scale generative models, diffusion-based approaches have emerged. Wang et al. (2023a) demonstrated a pretrained text-to-image diffusion can be adapted to artistic typography, and Yang et al. (2023) further leveraged glyph shapes to balance creativity with legibility.

Nevertheless, most previous studies are confined to 2D rendering, leaving 3D artistic font generation largely underexplored. Prior attempt DreamFont3D (Li et al., 2024) tried to leverage pre-trained 2D diffusion models to refine 3D NeRF volumes. However, Such a NeRF model struggles to achieve precise part-level control, as the implicit representation of NeRF lacks structural decomposition. Moreover, the optimization and rendering of NeRF are highly time-consuming and computationally expensive. Our work differs by leveraging the strong geometry priors of 3D Gaussians and successfully achieves structure-aware personalized 3D-AFG with explicit part-level style control. Our SplatFont3D attains much higher rendering efficiency with better generation quality over DreamFont3D, and our explicit part-level control is also more effective than the implicit one of NeRF.

2.2 Text-to-3D Generation

Early text-to-3D methods (Chen et al., 2018; Seo et al.) mainly relied on large-scale 3D assets, which are limited by dataset coverage and struggle with novel shapes. Recent advances (Lin et al., 2023) leveraged Score Distillation Sampling (SDS) (Poole et al., 2023) to optimize NeRF (Mildenhall et al., 2021) with pretrained 2D diffusion models, eliminating the need for real 3D data. Moreover, point- and Gaussian-based representations (Kerbl et al., 2023; Yi et al., 2024a; Chen et al., 2024b) have been proposed to improve optimization speed, memory efficiency, and structure control. However, existing text-to-3D models are primarily designed for general 3D objects, essentially making those models inapplicable due to the unique challenges of fonts (such as the strong semantics and shape constraints of characters). Nevertheless, 3D-AFG via text-to-3D models remains largely underexplored, especially for structure-aware 3D generation with part-level style control.

3 METHODOLOGY

3.1 OVERALL FRAMEWORK

Fig 1 illustrates the overall framework of our SplatFont3D, which aims to generate 3D customized artistic fonts with part-level style control. Our SplatFont3D consists of three parts: (1) glyph2Cloud for 3D Gaussian initialization, (2) 3D Gaussians optimization via score distillation sampling, and (3) structure-aware synthesis with part-level style control. We will introduce the details of each part.

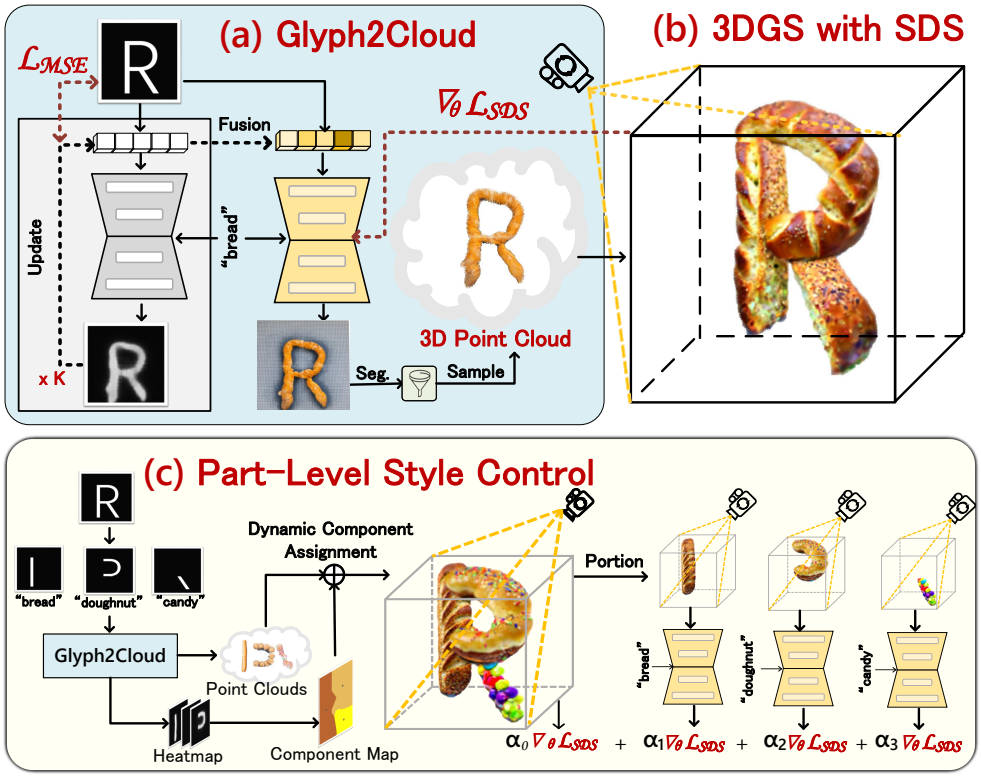

Figure 1: Overview of SplatFont3D for structure-aware 3D-AFG.

3.2 GLYPH2CLOUD FOR 3D GAUSSIAN INITIALIZATION

3D Gaussians require a well-initialized 3D point cloud for effective optimization, while it is challenging to directly generate accurate 3D font point clouds with pre-trained text-to-3D models. This is because these models are primarily trained on general objects and thus struggle with 3D font generation. To address this issue, we propose **Glyph2Cloud** to enable the creation of well-initialized 3D point clouds. The core idea is to leverage 2D printed glyphs as strong geometric priors and then utilize large pre-trained 2D diffusion models to generate 2D stylistic fonts that not only respect shape constraints but also preserve styles specified by textual prompts. Specifically,

2D Generation with Shape-Style Tradeoffs. We first adopt a pre-trained text-to-image diffusion model φ to reconstruct the object shapes of input images in latent space. Let z_p be the latent feature of the given printed glyph x_p, and y be the text prompts that specify the artistic styles, then the shape latent z_s is obtained as

$$z_s^t = \varphi(z_p^t, y, t), \tag{1}$$

under which we further introduce auxiliary reconstruction loss for shape constraints of z_s as

$$\mathcal{L}_{shape} = ||\mathcal{D}(z_s^t) - x_p||_1, \tag{2}$$

where $\mathcal{D}$ is the pretrained image decoder. After that, we perform a denoising intervention by injecting the shape latent z_s into the original z_g, which influences the generation of target 2D artistic fonts

x_g by enabling a trade-off between stylistic fidelity and shape preservation, i.e.,

$$\tilde{z}^t = \alpha \odot z_s^t + (1-\alpha) \odot z_p^t \quad , \quad t = T \dots T-K \tag{3}$$

$$\tilde{z}^t = \phi(z_p^t, y, t) \quad , \quad t = K \dots 0 \tag{4}$$

$$x_g = \mathcal{D}(\tilde{z}^0), \tag{5}$$

where ϕ is a pre-trained text-to-image diffusion model.

Sampling 3D Point Cloud for Gaussian Initialization. Empirically, this strategy often produces 2D stylistic fonts with clean backgrounds, facilitating the segmentation of foreground textures. Specifically, we adopt a segmentation model ClipSeg (Lüddecke & Ecker, 2022) ξ to predict the segmentation heatmap as

$$\mathcal{H}_g = \xi(x_g), \tag{6}$$

under which we can obtain the font foreground with simple pre-processing (such as thresholding). Subsequently, we perform uniform sampling on the segmented foreground fonts and project the sampled points into 3D space, thereby constructing 3D font point clouds for Gaussian initialization.

3.3 3D Gaussians Optimization via Score Distillation Sampling

3D Gaussian Splatting (3DGS). 3DGS represents an 3D font by a set of 3D Gaussians as

$$\mathcal{G} = \{(\mu_i, \Sigma_i, c_i, \alpha_i)\}_{i=1}^N, \tag{7}$$

whereμ_i and Σ_i denote the mean and covariance in 3D space, while c_i and α_i represent color and opacity. After projecting each Gaussian onto the image plane, the color of a pixel is rendered as

$$C(x) = \sum_{i=1}^{N} c_i \alpha_i \mathcal{N}(x|\tilde{u}_i, \tilde{\Sigma}_i) T_i, \tag{8}$$

where $\tilde{u}_i$ and $\tilde{\Sigma}_i$ are the projected mean and variance, $T_i = \prod_{j<i}(1-\alpha_i)$ denotes the accumulated transmittance for alpha blending, and $\mathcal{N}$ is the Gaussian kernel.

Score Distillation Sampling (SDS). To further leverage the priors of the pre-trained 2D diffusion model ϕ_x for 3D font generation, we accumulatively optimize 3D Guassians $\mathcal{G}$ through SDS , which projects the differentiable 3D representation from various viewpoints and makes the projected 2D images match the text conditions. Let the differentiable 3D Gaussians $\mathcal{G}$ transform parameters θ to render 2D images as $x = \mathcal{G}(\theta)$, the optimizing gradient is computed as

$$\nabla_\theta \mathcal{L}_{SDS}\left(\phi, x = \mathcal{G}(\theta)\right) \triangleq \mathbb{E}_{t,\epsilon}\left[w(t)(\hat{\epsilon}_\phi(z^t; y, t) - \epsilon)\right], \tag{9}$$

where $w(t)$ is a weight function, $\hat{\epsilon}_\phi(z^t; y, t)$ predicts the sampled noise $\hat{\epsilon}_\phi$ conditioned on the noisy latent z^t and the given text prompts y. By lifting 2D models into 3D, such an approach eliminates the need for real 3D data acquisition when optimizing 3D Gaussians.

3.4 Structure-Aware Synthesis with Part-Level Style Control

By leveraging the strong geometry priors of 3D Gaussians, we can achieve structure-aware 3D-AFG with precise part-level style control. Specifically,

Component-Wise Style Specification To enable part-level style control, we decompose the printed glyph x_p into M glyph components, where each component g_m is paired with the part-level style description y_m. Therefore, we obtain the part-level glyph-style annotations $\{(g_m, y_m)\}_{i=1}^M$. Therefore, we feed each component into the Glyph2Cloud to obtain the initial Gaussians $\{\mathcal{G}_m\}_{m=1}^M$, and we can also obtain the global font Gaussians $\mathcal{G}_0$ by composing components in the spatial space. Therefore, we can achieve part-level style control through component-wise SDS as

$$\nabla_\theta \mathcal{L}_{SDS} = \sum_{m=0}^{M} \lambda_i \nabla_\theta \mathcal{L}_{SDS}\left(\phi, \mathcal{G}_i(\theta)\right) \tag{10}$$

$$\triangleq \mathbb{E}_{t,\epsilon,m}\left[\lambda_i w(t)(\hat{\epsilon}_\phi(z_m^t; y_i, t) - \epsilon)\right], \tag{11}$$

where λ_m controls the importance of m-th part, and the pre-trained ϕ are shared for all Gaussians.

Dynamic Component Assignment Due to the dynamic optimization mechanism of 3DGS, Gaussian points may drift over iterations. As a result, points from different components can overlap and interfere with each other, ultimately degrading the overall quality of the generated fonts. To address this drift-induced component entanglement issue, we propose a dynamic component assignment strategy. Specifically, we leverage the 2D stylized font to obtain a component label map $\mathcal{M}$, where each pixel at 2D position p is assigned a component label as

$$\mathcal{M}(p) = \arg\max_m \left(\log(\mathcal{H}_m(p) + \delta) - \beta(||p - u_{\mathcal{H}}^m||_2)\right), \tag{12}$$

where $\mathcal{H}_m$ is the 2D heatmap of the m-th component (according to Eq. 6), $u_{\mathcal{H}}^m = \frac{\sum_{q \in g_m} q\mathcal{H}_m(q)}{\sum_{q \in g_m} \mathcal{H}_m(q)}$ is the centroid position of m-th component, and δ is an infinitesimal. Let $\tilde{u}_i$ be the projected 2D mean of the i-th Gaussian point, then we dynamically update the component label of each Gaussian point as $\mathcal{M}(\tilde{u}_i)$ and re-group the Gaussian points from each component $\mathcal{G}_m$ during 3DGS optimization. This eventually helps achieve more effective and efficient part-level style control for 3D-AFG.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUPS

Data Preparation. We constructed a collection of glyph–text pairs, including 10 printed digits, 26 uppercase English letters, and 8 Chinese characters. Style text prompts cover categories such as fruits, foods, and other general objects. For global style generation, printed glyphs are created from font library files. For part-level style control, the glyph is further divided into 2–3 components, each labeled with a style description. Other methods that only accept unified prompts can generate corresponding text prompts using GPT-4. Our data collection consists of 44 characters, each combined with 2 font styles and 2 modes (local or global), resulting in 1760 glyph–text pairs. Notably, we did not create or collect any realistic 3D fonts, since our SplatFont3D requires no realistic 3D data.

Implementation Details. Glyph2Cloud generated 2D images at 768×768 resolution, and the 3D fonts were rendered at 1024×1024. The model was optimized using Adam with a learning rate of 0.001 and the DDPM scheduler. We set $\lambda_0 = 0.01$ for global SDS and each local λ_i proportional to the region's area relative to the full glyph. Training was performed on an RTX 3090 GPU with PyTorch, with each font taking approximately 12 minutes for 3D-AFG with part-level style control.

Evaluation Metrics. The following metrics are utilized to thoroughly evaluate different models:

- *Semantic Consistency:* The **CLIP** score (Radford et al., 2021; Hessel et al., 2021) and **Alignment** (He et al., 2023) assessment measure the text-style consistency of the generated 3D fonts. They quantify the correlation between the text prompt and each 2D image rendered from different views.
- *Visual Quality and View Consistency:* The **Quality** (He et al., 2023; Xu et al., 2023), **V-LPIPS** (Zhang et al., 2018), and **V-CLIP** (Radford et al., 2021) measure the visual quality and view consistency of the generated 3D fonts. They quantify the correlation between the 2D images rendered from different views, where such view consistency reflects the visual quality of 3D objects.

More details of evaluation metrics can refer to Appendix A.3.

Competitors. As 3D-AFG remains underexplored, we can only compare classic text-to-3D models that can be adopted to this task, including **DreamFont3D** (Li et al., 2024), **DreamFusion** (Poole et al., 2023), **Latent-NeRF** (Metzer et al., 2023), **MVDream** (Shi et al., 2023), **Wonder3D** (Long et al., 2024; Rombach et al., 2022), **Fantasia3D** (Chen et al., 2023), **GSGEN** (Chen et al., 2024b), **GaussianDreamer** (Yi et al., 2024a), **GaussianDreamerPro** (Yi et al., 2024b).

Synthesis Tasks. We thoroughly evaluated different models under the following scenarios:

- *Global Style Generation.* The model produces each 3D font with a consistent global style. Input is either the glyph paired with a single style description or a corresponding unified text prompt.
- *Part-Level Style Control.* The model generates 3D fonts with distinct styles applied to different components of each glyph. Inputs consist of either multiple component images, each with a single-style description, or a unified text prompt specifying the character with part-level styles.

4.2 COMPARISON WITH SOTA METHODS

To demonstrate the effectiveness of our method, we compared our SplatFont3D with existing text-to-3D models through both quantitative and qualitative analyses regarding the generation performance.

Method	Global Style Generation				
	CLIP↑	Alignment↑	Quality↑	V-LPIPS↓	V-CLIP↑
Wonder3D	0.64	3.09	25.28	0.51	0.74
MVDream	0.70	2.81	29.77	0.36	0.89
Latent-NeRF	0.64	2.34	17.12	0.19	0.92
GsGen	0.66	3.57	37.17	0.31	0.92
DreamFusion	0.60	3.60	17.61	**0.16**	0.91
GaussianDreamer	0.71	3.62	40.36	0.19	0.92
GaussianDreamerPro	0.76	2.91	40.90	0.35	0.85
Fantasia3D	0.63	3.24	36.58	0.36	0.91
DreamFont3D	**0.82**	**4.38**	35.62	0.19	**0.96**
SplatFont3D (Ours)	0.80	4.02	**53.11**	0.18	0.93
	Part-Level Style Control				
Wonder3D	0.65	3.59	22.87	0.55	0.75
MVDream	0.65	2.81	22.10	0.26	0.91
Latent-NeRF	0.56	3.17	15.50	0.20	0.93
GsGen	0.68	3.74	35.21	0.34	0.92
DreamFusion	0.62	2.57	20.37	0.21	0.92
GaussianDreamer	0.70	3.70	33.74	0.21	0.93
GaussianDreamerPro	0.79	2.42	34.34	0.31	0.89
Fantasia3D	0.65	3.75	32.10	0.36	0.91
DreamFont3D	0.81	3.22	33.82	0.21	**0.95**
SplatFont3D (Ours)	**0.84**	**4.14**	**48.89**	**0.19**	0.92
	Global Generation + Part-Level Control				
Wonder3D	0.65	3.34	24.08	0.53	0.74
MVDream	0.66	2.81	25.89	0.31	0.90
Latent-NeRF	0.66	2.81	25.89	0.31	0.90
GsGen	0.67	3.65	36.19	0.32	0.92
DreamFusion	0.62	3.09	18.99	0.19	0.91
GaussianDreamer	0.71	3.66	37.05	0.20	0.92
GaussianDreamerPro	0.77	2.67	37.62	0.33	0.87
Fantasia3D	0.64	3.50	34.34	0.36	0.91
DreamFont3D	0.81	3.80	34.72	0.20	**0.96**
SplatFont3D (Ours)	**0.82**	**4.08**	**51.00**	**0.18**	0.93

Table 1: Quantitative comparisons of different methods for 3D-AFG under different settings.

Quantitative Results Table 1 reports the quantitative results of different methods for 3D-AFG under different settings, including "Global Style Generation", "Part-Level-Style Control", and the combination of the two. We can observe that our SplatFont3D achieves competing performance with existing 3D models for global style generation, while largely outperforming previous 3D models for part-level style control, especially in terms of style-text consistency (e.g., Alignment scores) and visual quality (e.g., Quality and P-Lpips scores). Overall, quantitative comparisons demonstrated that our SplatFont3D achieves the SoTA performance for 3D-AFG, especially for structure-aware generation with part-level style control.

Qualitative Results Fig 2 illustrates the qualitative comparison between our SplatFont3D and the current SoTA text-to-3D models for 3D-AFG with "Global Style Generation" and "Part-Level Style Control". We can observe that it is inapplicable to directly adopt the existing 3D models for 3D-AFG, due to the unique challenges of 3D-AFG over 3D general object synthesis. Although DreamFont3D can generate recognizable 3D fonts of digits and letters, it struggled to synthesize 3D artistic fonts of complex structures (i.e., Chinese characters). Instead, our SplatFont3D exhibits more accurate font effects with more precise locations and better recognizability, even for Chinese characters of complex structures.

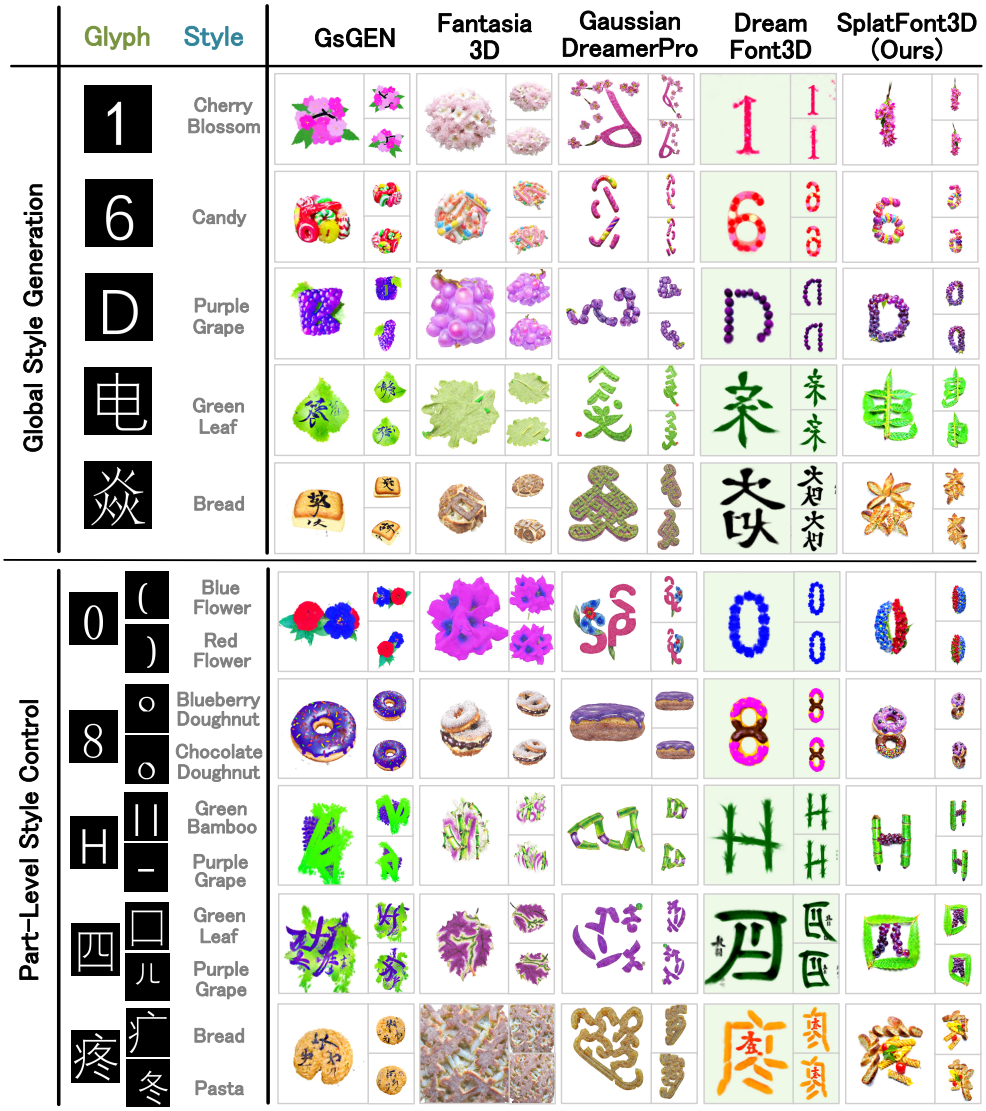

Figure 2: Qualitative comparisons of global style generation and part-level style control.

Rendering Efficiency Comparison Fig. 3 reports the rendering efficiency comparison of different methods for 3D-AFG on a RTX-3090 GPU. It can be observed that our SplatFont3D achieves a clear advantage in rendering speed over other 3D models. Beyond the inherent efficiency of 3DGS, this improvement is attributed to our two key designs: (1) Glyph2Cloud, which provides the well-initialized Gaussians for optimization from a better starting point, and (2) Dynamic Component Assignment, which prevents Gaussian point drifting during the optimization process. Together, these enable SplatFont3D to achieve faster and more stable rendering.

4.3 Ablation Study

ID	G2C	DCA	Part-Level Style Control				
			CLIP↑	Alignment↑	Quality↑	V-LPIPS↓	V-CLIP↑
1	×	×	0.73	<u>3.42</u>	36.85	0.26	0.87
2	✓	×	0.71	3.05	<u>43.50</u>	0.27	<u>0.89</u>
3	×	✓	<u>0.77</u>	3.30	41.38	<u>0.25</u>	0.86
4	✓	✓	**0.83**	**3.94**	**47.36**	**0.21**	**0.92**

Table 2: Ablation results on framework components.

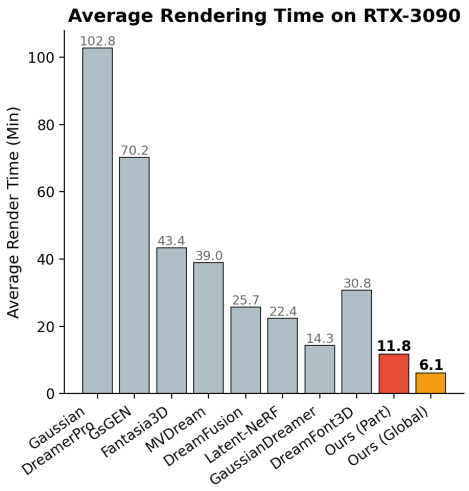

Figure 3: Rendering time comparison.

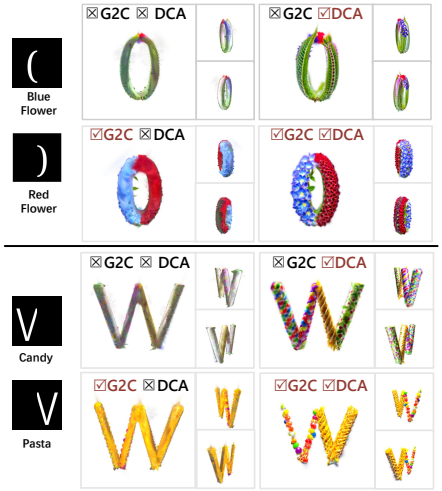

Figure 4: Qualitative ablation results.

Quantitative and Quantitative Ablation Results To demonstrate the effectiveness of Glyph2Cloud (G2C) and Dynamic Component Assignment (DCA) of SplatFont3D, we presented quantitative ablation results in Table 2 and qualitative ablation results in Fig. 4. Results indicate that both modules significantly enhance structure-aware 3D-AFG with precise part-level style control.

Glyph2Cloud for Shape-Style Tradeoffs. As shown in Fig. 5, we demonstrated that our Clyph2Cloud can achieve customized shape-style tradeoffs for 3D-AFG. By dynamically adjusting the hyperparameters K and α in Eq. (3), it enables controllable shape-style tradeoffs.

Figure 5: Glyph2Cloud for shape-style tradeoffs: (a) 2D results and (b) the final 3D fonts.

5 Conclusion

Most existing studies are limited to generating 2D artistic fonts, leaving the 3D artistic font generation largely underexplored. In this paper, we presented SplatFont3D, a structure-aware text-to-3D artistic font generation framework that enables fine-grained part-level style control. Specifically, our Glyph2Cloud module progressively refines 2D glyphs while preserving semantic consistency, producing well-initialized 3D point clouds for Gaussian-based modeling. By integrating 2D diffusion priors with 3D Gaussian geometry and employing a dynamic component assignment strategy, SplatFont3D effectively resolves drift-induced component entanglement, achieving explicit and controllable part-level styling without requiring real 3D font data. Extensive experiments show that our method outperforms existing text-to-3D approaches in style–text consistency, visual quality, and rendering efficiency, demonstrating its effectiveness and potential for immersive 3D applications.

Ethics Statement

This work introduces a method for structure-aware 3D artistic font generation with part-level style control, intended for applications in digital design, creativity, and accessibility. All pretrained models used are publicly available, and all data collections are synthetically generated, and no personal or sensitive information is involved. While the technique could potentially be misused to imitate proprietary fonts, our contribution is intended solely for research and creative purposes, and we encourage responsible use. We acknowledge limitations in stylistic diversity and the environmental cost of training, and we have aimed to minimize computational overhead where possible.

Reproducibility statement

We have taken steps to ensure the reproducibility of our work. The full model architecture, training procedure, and hyperparameters are described in Section 3. All evaluation metrics are formally defined in Section 4 and Appendix A.3, and all used pre-trained 2D diffusion models are publicly available. We only use the synthetically generated data collections, where the data generation processes are well described in Section 4.1. We also disclose the use of large language models (LLMs) in the Appendix, including what LLMs are used for metric evaluation and prompt construction in experiments. All used LLMs are publicly available. Moreover, comprehensive experimental details, including training configurations, evaluation scenarios, and evaluation metrics, are provided in Section 4.

References

Kevin Chen, Christopher B Choy, Manolis Savva, Angel X Chang, Thomas Funkhouser, and Silvio Savarese. Text2shape: Generating shapes from natural language by learning joint embeddings. In *Asian conference on computer vision*, pp. 100–116. Springer, 2018.

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*, pp. 22246–22256, 2023.

Xinping Chen, Xiao Ke, and Wenzhong Guo. If-font: Ideographic description sequence-following font generation. *Advances in Neural Information Processing Systems*, 37:14177–14199, 2024a.

Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 21401–21412, 2024b.

Yue Gao, Yuan Guo, Zhouhui Lian, Yingmin Tang, and Jianguo Xiao. Artistic glyph image synthesis via one-stage few-shot learning. *ACM Transactions on Graphics (ToG)*, 38(6):1–12, 2019.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Hideaki Hayashi, Kohtaro Abe, and Seiichi Uchida. Glyphgan: Style-consistent font generation based on generative adversarial networks. *Knowledge-Based Systems*, 186:104927, 2019.

Yuze He, Yushi Bai, Matthieu Lin, Wang Zhao, Yubin Hu, Jenny Sheng, Ran Yi, Juanzi Li, and Yong-Jin Liu. T^3 bench: Benchmarking current progress in text-to-3d generation. *arXiv preprint arXiv:2310.02977*, 2023.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021.

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

Tianyu Huang, Yihan Zeng, Bowen Dong, Hang Xu, Songcen Xu, Rynson W. H. Lau, and Wangmeng Zuo. Textfield3d: Towards enhancing open-vocabulary 3d generation with noisy text fields. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=WOiOzHG2zD.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023a.

Xiang Li, Lei Wu, Xu Chen, Lei Meng, and Xiangxu Meng. Dse-net: Artistic font image synthesis via disentangled style encoding. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2022.

Xiang Li, Lei Wu, Changshuo Wang, Lei Meng, and Xiangxu Meng. Compositional zero-shot artistic font synthesis. In *IJCAI*, pp. 1098–1106, 2023b.

Xiang Li, Lei Meng, Lei Wu, Manyi Li, and Xiangxu Meng. Dreamfont3d: personalized text-to-3d artistic font generation. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024.

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*, pp. 300–309, 2023.

Zhen Liu, Yao Feng, Michael J. Black, Derek Nowrouzezahrai, Liam Paull, and Weiyang Liu. Meshdiffusion: Score-based generative 3d mesh modeling. In *The Eleventh International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=0cpM2ApF9p6.

Zhengzhe Liu, Peng Dai, Ruihui Li, XIAOJUAN QI, and Chi-Wing Fu. ISS: Image as stepping stone for text-guided 3d shape generation. In *The Eleventh International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=GMRodZ8OlVr.

Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9970–9980, 2024.

Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7086–7096, June 2022.

Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12663–12673, 2023.

Yalin Miao, Huanhuan Jia, and Kaixu Tang. Artistic font generation network combining font style and glyph structure discriminators. *Multimedia Tools and Applications*, 83(8):21883–21903, 2024.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

Xinzhi Mu, Li Chen, Bohan Chen, Shuyang Gu, Jianmin Bao, Dong Chen, Ji Li, and Yuhui Yuan. Fontstudio: shape-adaptive diffusion model for coherent and consistent font effect generation. In *European Conference on Computer Vision*, pp. 305–322. Springer, 2024.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

Song Park, Sanghyuk Chun, Junbum Cha, Bado Lee, and Hyunjung Shim. Few-shot font generation with localized style representations and factorization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 2393–2402, 2021.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023.

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*, pp. 8748–8763. PmLR, 2021.

Minsi Ren, Yan-Ming Zhang, and yi chen. Decoupling layout from glyph in online chinese handwriting generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=DhHIw9Nbl1.

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*, pp. 10684–10695, 2022.

Junyoung Seo, Susung Hong, Wooseok Jang, Min-Seop Kwak, Hyeonsu Kim, Doyup Lee, and Seungryong Kim. Retrieval-augmented text-to-3d generation.

Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Yuchen Tian. zi2zi: Master chinese calligraphy with conditional adversarial networks. *Internet] https://github. com/kaonashi-tyc/zi2zi*, 3(2), 2017.

Changshuo Wang, Lei Wu, Xiaole Liu, Xiang Li, Lei Meng, and Xiangxu Meng. Anything to glyph: artistic font synthesis via text-to-image diffusion model. In *SIGGRAPH Asia 2023 Conference Papers*, pp. 1–11, 2023a.

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. *Advances in neural information processing systems*, 36:8406–8441, 2023b.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Yukang Yang, Dongnan Gui, Yuhui Yuan, Weicong Liang, Haisong Ding, Han Hu, and Kai Chen. Glyphcontrol: Glyph conditional control for visual text generation. *Advances in Neural Information Processing Systems*, 36:44050–44066, 2023.

Taoran Yi, Jiemin Fang, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6796–6807, 2024a.

Taoran Yi, Jiemin Fang, Zanwei Zhou, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Xinggang Wang, and Qi Tian. Gaussiandreamerpro: Text to manipulable 3d gaussians with highly enhanced quality. *arXiv preprint arXiv:2406.18462*, 2024b.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

A APPENDIX

THE USE OF LARGE LANGUAGE MODELS (LLMS)

We disclose that large language models (LLMs) were used in three limited contexts:

1. **Language Polishing** – We solely used LLMs to polish the writing, specifically for spelling and grammar checking.
2. **Evaluation for Alignment Assessment** – When computing the *Alignment Assessment* (see Appendix A.3), we employed BLIP-2 to generate captions for images from each viewpoint, and then used GPT-4 to evaluate the consistency between the generated captions and the corresponding 2D view images.
3. **Prompt Construction** – for text-to-3D models that only accept text prompts as input, we used GPT-4 to generate text prompts depending on the given text–glyph pairs.

Therefore, we confirm that LLMs did not contribute to the research ideation, methodology, experimental design, analysis, or substantive writing of this paper.

A.1 MORE QUALITATIVE RESULTS OF OUR SPLATFONT3D

As shown in Fig. 6, we also provide more qualitative results of our SplatFont3D for structure-aware 3D-AFG with part-level style control. Experimental results demonstrate the effectiveness of our approach in achieving both structural fidelity and flexible style manipulation for structure-aware 3D-AFG with customized part-level style control.

Figure 6: Qualitative results of our splatFont3D for structure-aware 3D-AFG.

A.2 MORE QUALITATIVE COMPARISONS OF DIFFERENT MODELS

To further demonstrate the effectiveness of our method for 3D-AFG, we provide a more thorough qualitative comparison between our SplatFont3D and existing text-to-3D models regarding the generation performance, including the **Global Style Generation in Fig. 7** and **Part-Level Style Control in Fig. 8**. These comparisons show that our SplatFont3D produces more faithful global styles and provides finer part-level control than existing approaches, achieving more consistent global styles and finer-grained part-level control for 3D-AFG.

Figure 7: Qualitative comparison of different methods for part-level style control.

A.3 Details of Evaluation Metrics

- **CLIP:** Measures the semantic fidelity of generated glyphs by encoding multiple rendered views with CLIP (Radford et al., 2021; Hessel et al., 2021) and computing the cosine similarity to the corresponding textual prompt, then averaging across views to obtain a robust multi-view score.
- **Alignment Assessment:** Evaluates higher-level semantic correspondence by generating captions for each view with BLIP-2 (Li et al., 2023a), consolidating them into a single summary using GPT-4, and then prompting the model to rate the alignment between the summary and the original prompt on a five-point scale.
- **Quality Assessment:** Evaluates the visual fidelity of generated glyphs by applying ImageReward (Xu et al., 2023) to multi-view renderings conditioned on the input prompt. To reduce view-to-view noise, each score is smoothed using a local neighborhood average over adjacent views, producing a more consistent assessment of overall image quality.

Figure 8: Qualitative comparison of different methods for global style generation.

- **V-LPIPS:** Measures multi-view perceptual consistency of generated artistic glyphs by computing LPIPS (Zhang et al., 2018) between adjacent rendered views and averaging the results. This metric captures how smoothly the glyph's appearance transitions across viewpoints, reflecting both structural and stylistic coherence.
- **V-CLIP:** Evaluates multi-view semantic consistency of generated artistic glyphs by computing the cosine similarity between CLIP embeddings of adjacent views and averaging the results, capturing how consistently the glyph preserves the intended semantics across viewpoints.