StyleTex: Style Image-Guided Texture Generation for 3D Models

ZHIYU XIE*, State Key Lab of CAD&CG, Zhejiang University, China
YUQING ZHANG*, State Key Lab of CAD&CG, Zhejiang University, China
XIANGJUN TANG, State Key Lab of CAD&CG, Zhejiang University, China
YIQIAN WU, State Key Lab of CAD&CG, Zhejiang University, China
DEHAN CHEN, State Key Lab of CAD&CG, Zhejiang University, China
GONGSHENG LI, Zhejiang University, China
XIAOGANG JIN†, State Key Lab of CAD&CG, Zhejiang University, China

Fig. 1. StyleTex is capable of generating visually compelling and harmonious stylized textures for a given scene. For each mesh in the 3D scene, StyleTex utilizes the untextured mesh, a single reference image, and a text prompt describing the mesh and desired style as inputs to generate a stylized texture. The generated textures preserve the style of the reference image while ensuring consistency with both the text prompts and the intrinsic details of the given 3D mesh. At the bottom, we present the rendered output for the provided 3D scene with the generated texture.

Style-guided texture generation aims to generate a texture that is harmonious with both the style of the reference image and the geometry of the input mesh, given a reference style image and a 3D mesh with its text description. Although diffusion-based 3D texture generation methods, such as distillation sampling, have numerous promising applications in stylized games and films, it requires addressing two challenges: 1) decouple style and content completely from the reference image for 3D models, and 2) align the generated texture with the color tone, style of the reference image, and the given text prompt. To this end, we introduce StyleTex, an innovative diffusion-model-based framework for creating stylized textures for 3D models. Our key insight is to decouple style information from the reference image while disregarding content in diffusion-based distillation sampling. Specifically, given a reference image, we first decompose its style feature from the image CLIP embedding by subtracting the embedding's orthogonal projection in the direction of the content feature, which is represented by a text CLIP embedding. Our novel approach to disentangling the reference image's style and content information allows us to generate distinct style and content features. We then inject the style feature into the cross-attention mechanism to incorporate it into the generation process, while utilizing the content feature as a negative prompt to further dissociate content information. Finally, we incorporate these strategies into StyleTex to obtain stylized textures. We utilize Interval Score Matching to address over-smoothness and over-saturation, in combination with a geometry-aware ControlNet that ensures consistent geometry throughout the generative process. The

*Equal contribution
†Corresponding author.

Authors' addresses: Zhiyu Xie, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, China, xiezhiyu@zju.edu.cn; Yuqing Zhang, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, China, 3180102110@zju.edu.cn; Xiangjun Tang, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, China, xiangjun.tang@outlook.com; Yiqian Wu, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, China, onethousand1250@gmail.com; Dehan Chen, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, China, cdh573885@outlook.com; Gongsheng Li, Zhejiang University, Hangzhou, Zhejiang, China, ligongshengzju@foxmail.com; Xiaogang Jin, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, China, jin@cad.zju.edu.cn.

ACM 0730-0301/2024/12-ART212
https://doi.org/10.1145/3687931

resulting textures generated by StyleTex retain the style of the reference image, while also aligning with the text prompts and intrinsic details of the given 3D mesh. Quantitative and qualitative experiments show that our method outperforms existing baseline methods by a significant margin.

CCS Concepts: • **Computing methodologies** → **Rendering**.

Additional Key Words and Phrases: Image-guided texturing, Stylization

ACM Reference Format:
Zhiyu Xie, Yuqing Zhang, Xiangjun Tang, Yiqian Wu, Dehan Chen, Gongsheng Li, and Xiaogang Jin. 2024. StyleTex: Style Image-Guided Texture Generation for 3D Models. *ACM Trans. Graph.* 43, 6, Article 212 (December 2024), 13 pages. https://doi.org/10.1145/3687931

1 INTRODUCTION

We investigate an under-explored generation problem: style image-guided texture synthesis, which is crucial in computer vision and graphics, facilitating the creation of visually compelling and immersive digital environments in games and films. The generated texture needs to be harmonious with both the 3D shape and style of the reference image, which requires the texture to align with the geometry while conveying a consistent style from different views.

Existing research mostly investigates the above two requirements separately. In 2D style-image generation methods, the style is conveyed by separating it from the reference image and incorporating it into the final output, which usually involves fine-tuning [Gal et al. 2023; Hu et al. 2022; Ruiz et al. 2023] the diffusion model to be a stylized image generator or adjusting the hidden layers of the diffusion model with the extracted style features [He et al. 2024; Hertz et al. 2024; Jeong et al. 2024; Voynov et al. 2023; Wang et al. 2024]. In parallel, 3D texture can be generated by iteratively inpainting [Chen et al. 2023c; Richardson et al. 2023] or image synthesis with multi-view consistency [Cao et al. 2023; Gao et al. 2024; Liu et al. 2023a; Wu et al. 2024]. More recently, distillation methods such as score distillation sampling [Chen et al. 2023a; Metzer et al. 2023; Youwang et al. 2023] have also proven their superior effectiveness in synthesizing 3D consistent textures. Compared to the direct generation of textures, distillation methods are capable of achieving better view and global style consistency while avoiding local seam problems.

Despite the progress in these two distinct areas, incorporating the desired style into texture generation is not straightforward. One possible solution is to combine the distillation method with a diffusion distribution aligned with the reference image's style. However, this leads to two challenges: 1) decoupling the style and content from the reference image entirely, and 2) preserving the color tone. Firstly, the ambiguity between style and content from different views complicates the decoupling process. In 2D domains, separating style and content within a single viewpoint may succeed in most situations. However, in 3D domains, failure to effectively decouple style from any single viewpoint can result in inaccurate style and unintended content leakage in the final texture. Thus, the generation of stylized textures in 3D domains requires a robust method for disentangling style and content. Secondly, distillation methods may result in over-saturation and over-smoothing within the generated textures, leading to color shifts and a lack of details, hindering the accurate reflection of the intended style.

To overcome these challenges, we propose **StyleTex**, a diffusion-model-based pipeline to generate style textures under the guidance of a single image. Our key insight is to extract the style information from the reference image while disregarding the content information. Inspired by the multi-modal applications of the CLIP space, we propose to represent the content of the reference image as the CLIP embedding of its corresponding text prompt. A naive method to discard the content from the reference image in InstantStyle [Wang et al. 2024] is to drive the reference image embedding in the same CLIP space toward the opposite direction of the content embedding. However, the slight misalignment between the content embedding and the real content information of the image may cause undesirable image embedding alerting, which results in unclean content information remaining or color tone changing. To address this, we remove the content information from the reference image embedding by decomposing its CLIP embedding into two separate orthogonal features. One of these features aligns with the content embedding and encodes most of the content information of the reference image. We retain only the remaining feature, which predominantly relates to the style, to refine our diffusion model. To this end, we explicitly incorporate the style-relevant feature through the cross-attention mechanism, which also serves as a color tone guidance that can prevent unintentional color tone changing during the distillation process. Furthermore, we incorporate the content embedding as a negative prompt to further dissociate content information. We integrate the aforementioned strategies into StyleTex to generate stylized textures and utilize Interval Score Matching (ISM) [Liang et al. 2024] to further tackle the issue of over-smoothness. Moreover, we utilize a geometry-aware ControlNet to ensure geometric consistency throughout the generative process.

In summary, our work makes the following major contributions:

- A diffusion-model-based pipeline to generate style textures under the guidance of a single image, enabling the automatic creation of diverse stylized virtual environments.
- A novel style decoupling and injection strategy that effectively guides stylization while addressing issues of content leakage and style deviation in texture generation.

2 RELATED WORK

2.1 Image guided stylization

Given a reference image, image-guided stylization aims to synthesize a new image that shares the same style as the reference image while demonstrating the intended content. Early methods [Chen and Schmidt 2016; Gatys et al. 2016; Gu et al. 2018] alter the style of an image while preserving its content by solving a slow optimization. The following methods propose to represent the style by a neural network [An et al. 2021; Chen et al. 2017; Dumoulin et al. 2017; Johnson et al. 2016; Ulyanov et al. 2016; Zhang and Dana 2019], or by the statistics of the hidden features of a network [Huang and Belongie 2017; Kolkin et al. 2022; Kotovenko et al. 2019; Li et al. 2017; Park and Lee 2019], enabling stylization by a single-step inference of the network. With the development of the text-to-image diffusion model [Rombach et al. 2022], fine-tuning the diffusion model [Chen et al. 2023b; Frenkel et al. 2024; Gal et al. 2023; Hu et al. 2022; Ruiz et al. 2023; Shah et al. 2023; Sohn et al. 2024] yields a stylized image

generator but requires time-consuming training. Based on the existing style representations, modifying the structure of the diffusion model [Hertz et al. 2024; Jeong et al. 2024; Zhang et al. 2023a] and utilizing other adapter-based methods [Qi et al. 2024; Wang et al. 2024, 2023; Ye et al. 2023] allows for the incorporation of desired styles without training.

Image-guided 3D stylization can be analogous to the 2D methods but replaces the 2D image with the 3D representations, such as point clouds [Huang et al. 2021; Mu et al. 2022], mesh [Höllein et al. 2022; Kato et al. 2018; Yin et al. 2021], NeRF [Huang et al. 2022; Kolkin et al. 2022; Liu et al. 2023b; Nguyen-Phuoc et al. 2022; Zhang et al. 2022] or 3D Gaussian [Zhang et al. 2024a]. However, establishing style consistency over multiple views in 3D space has not been fully explored, leading to artifacts such as content leakage.

2.2 Text/Image-guided Texture Generation

Automatically generating textures over 3D surfaces has garnered widespread attention and important applications. While training the texture generation network on a small dataset [Chen et al. 2022; Siddiqui et al. 2022] aids in learning a stylized distribution, it also restricts the network to a particular texture category. Text-to-image diffusion [Rombach et al. 2022] incorporates a strong 2D image prior that represents a real image distribution, offering robust guidance for text-driven texture generation. For instance, TEXTure [Richardson et al. 2023] and Text2Tex [Chen et al. 2023c] employ the diffusion model to iteratively inpaint the geometry from different viewpoints. However, the 2D diffusion model lacks an understanding of 3D shape and multi-view color consistency, leading to blurry and low-quality texture results. To maintain 3D consistency, a possible way is to employ a 3D consistent prior [Chen et al. 2023a; Guo et al. 2023; Le et al. 2023; Metzer et al. 2023], such as applying the score distillation sampling using a geometry-conditioned diffusion model. In addition, methods such as SyncMVD [Liu et al. 2023a], TexRO [Wu et al. 2024] and GensisTex [Gao et al. 2024] are also able to maintain the 3D consistency by explicitly projecting the intermediate results of each denoising step into a consistent texture space.

Instead of employing diffusion models designed for a real-image distribution, another viable alternative could be to fine-tune a diffusion model to learn a UV space texture distribution [Cheskidova et al. 2023; Liu et al. 2024; Zeng et al. 2023a]. This approach can significantly accelerate the generation process, but the results may be heavily impacted by the quality of the UV mapping.

Unlike text-driven texture generation, image-guided approaches require interpreting the style of an image and hence cannot simply rely on the pretrained text-to-image model. In addition to text-guided texture generation, there have also been attempts in image-guided texture generation. TEXTure [Richardson et al. 2023] employs textual inversion [Gal et al. 2023] to capture the style and structural features of reference images, while Texturedreamer [Yeh et al. 2024] uses Dreambooth [Ruiz et al. 2023] to fine-tune the Stable Diffusion and then applies the personalized model in the geometry-aware score distillation. However, these methods often require a fine-tuning process and cannot exclude the content information of the reference images. In contrast, our method is dedicated to decoupling the style and content information and generating textures consistent with the style of the reference images. As a result, the content and details of the textures are consistent with the textual prompts and the model's geometry, all without the need for an additional training process.

3 METHOD

Given an untextured mesh, a textual prompt, and a reference image, our goal is to generate textures consistent with the image style while aligning the content of the textures with both the textual prompt and the geometry of the model. In Sec. 3.1, we introduce the prior knowledge relevant to our method, including the diffusion denoising process and interval score matching (ISM) loss. In Sec. 3.2, we present our pipeline for generating stylized textures. In Sec. 3.3, we present our approach to style infusion, which includes transformer layer style injection and content and style disentanglement.

3.1 Preliminary

When it comes to text-to-3D, numerous approaches have been developed to optimize 3D representations by distilling 2D diffusion models, using techniques like score distillation sampling (SDS) [Poole et al. 2022]. The optimization goal of SDS is to make the renderings of 3D representations align with the image distribution in a pre-trained text-to-image diffusion model. At each iteration, the differentiable rendering function g renders the trainable paramaters θ from camera c, getting the rendered image x_0. After that, x_0 undergoes a noise addition process, resulting in $x_t \sim \mathcal{N}\left(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\boldsymbol{I}\right)$. With a text prompt y, a pre-trained 2D diffusion model is utilized to predict the corresponding noise. The gradient of the SDS loss with respect to the 3D representation is determined as follows:

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\theta) \approx \mathbb{E}_{t,\epsilon,c}\left[\omega(t)\left(\epsilon_\phi\left(x_t, t, y\right) - \epsilon\right)\frac{\partial g(\theta, c)}{\partial \theta}\right], \quad (1)$$

where $\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ is the ground truth denoising direction of x_t at timestep t, $\epsilon_\phi\left(x_t, t, y\right)$ is the predicted denoising direction [Liang et al. 2024] under the given condition y, and $\omega(t)$ denotes a weighting function that absorbs the constant $\alpha_t \mathbf{I} = \partial x_t / \partial x_0$. This equation can be rewritten as:

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\theta) = \mathbb{E}_{t,\epsilon,c}\left[\frac{\omega(t)}{\gamma(t)}\left(x_0 - \hat{x}_0^t\right)\frac{\partial g(\theta, c)}{\partial \theta}\right], \quad (2)$$

where $\gamma(t) = \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}$, and $\hat{x}_0^t = \frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\phi(x_t, t, y)}{\sqrt{\bar{\alpha}_t}}$ is the pseudo-GT [Liang et al. 2024] estimated by the single-step Diffusion Probabilistic Model (DDPM) [Ho et al. 2020] .

Based on SDS, Interval Score Matching (ISM) [Liang et al. 2024] generates a reversible diffusion trajectory by adding noise to x_0 through Denoising Diffusion Implicit Models (DDIM) [Song et al. 2020] inversion, and employing multi-step DDIM denoising process. This helps to achieve a more consistent and higher-quality $\hat{x}_0^t$. This process of noise addition and subsequent denoising facilitates the neutralization of a series of neighboring interval scores with opposing scales, resulting in the formulation of the ISM loss:

$$\nabla_\theta \mathcal{L}_{\text{ISM}}(\theta) = \mathbb{E}_{t,c}\left[\omega(t)\delta(x_t, x_{t-1}, t, t-1)\frac{\partial g(\theta, c)}{\partial \theta}\right], \quad (3)$$

$$\delta(x_t, x_{t-1}, t, t-1) = \epsilon_\phi\left(x_t, t, y\right) - \epsilon_\phi\left(x_{t-1}, t-1\right). \quad (4)$$

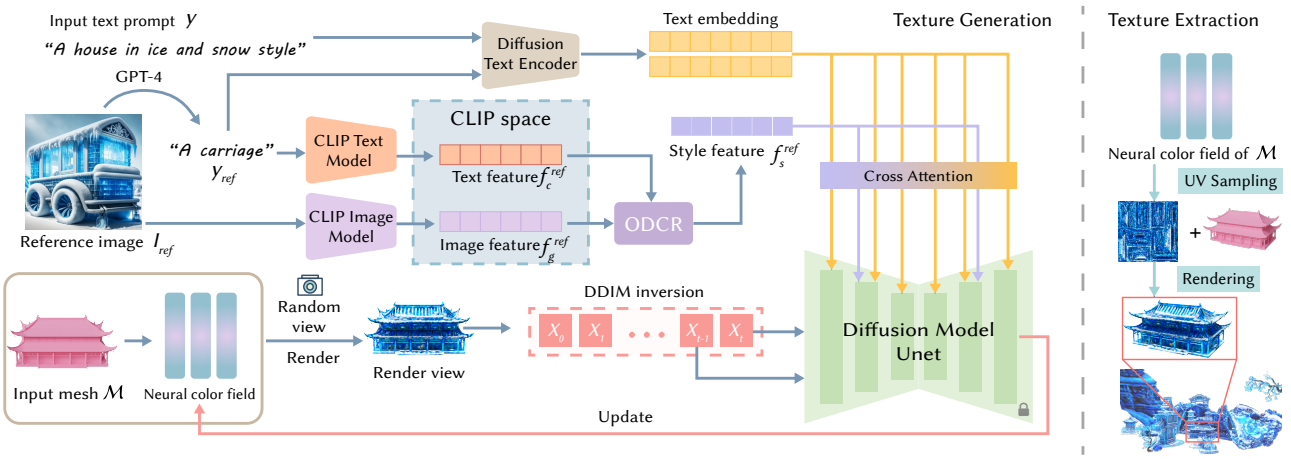


Fig. 2. **Overview of our pipeline.** StyleTex's inputs include a reference style image I_{ref}, a text prompt y, and an untextured 3D mesh $\mathcal{M}$. During training, we utilize our innovative ODCR method (described in Sec. 3.3) to extract a content-unrelated style feature, f_s^{ref}, from the reference image. The style feature and text embeddings are fed into the Unet to guide the optimization of the texture field. During inference, texture maps can be sampled from the texture field and directly employed in downstream game or film production, enabling the creation of stylized digital environments.

Compared to the SDS loss, ISM enhances the generation of 3D results by replacing the single-step DDPM with the multi-step DDIM, resulting in outputs with richer details. This approach effectively mitigates the issues of over-smoothness and blurriness in the results and notably accelerates the convergence rate. In this paper, we adopt ISM as the basis of our method to achieve more robust results.

Classifier-free guidance (CFG) [Ho and Salimans 2021] is also employed in diffusion models with a guidance weight λ_{cfg} to direct the unconditional score distribution to the conditional one. Specifically, $\delta(x_t, x_{t-1}, t, t-1)$ with CFG is expressed as:

$$\begin{aligned}\delta(x_t, x_{t-1}; t, t-1) &= \epsilon_\phi\left(x_t; t\right) - \epsilon_\phi\left(x_{t-1}; t-1\right) \\ &+ \lambda_{cfg}\left(\epsilon_\phi\left(x_t; t, y\right) - \epsilon_\phi\left(x_t; t\right)\right).\end{aligned} \tag{5}$$

Inspired by CFG, we employ a similar formulation in our proposed method to direct the unstylized score distribution to a stylized one, thereby achieving stylization.

3.2 Style-guided Texture Generation Pipeline

Our stylized texture generation pipeline is depicted in Fig. 2. The input encompasses an untextured 3D mesh denoted as $\mathcal{M}$, a reference image I_{ref} providing the style. We utilize GPT-4 to extract a text prompt y from the reference image I_{ref}, which characterizes the desired style and content, and a text prompt y_{ref} that describes the content of the reference image. Instead of directly optimizing the texture map in 2D space, we optimize a neural color field $\Gamma_\theta(p) = c$, where $p \in \mathcal{R}^3$ is the surface position of the 3D mesh and $c \in \mathcal{R}^3$ denotes the color. We represent the neural field by the hash-grid proposed by [Müller et al. 2022]. After optimization, the texture map can be sampled from the neural field, which is detailed in the supplement material.

At each iteration, in addition to rendering the image x_0, we render the depth and normal maps indicating the geometric information, which are incorporated into the optimization by a geometry-aware ControlNet [Zhang et al. 2023b] to achieve geometry consistency.

Besides, inspired by ISM [Liang et al. 2024], we incorporate a high-quality noise estimation method of ControlNet. Instead of simply sampling from a Gaussian distribution, we generate the noised x_t by utilizing DDIM inversion to achieve superior noise estimation.

Specifically, the parameters θ of the neural field Γ_θ are optimized by our novel style-guided loss. The gradient of our loss is:

$$\nabla_\theta \mathcal{L}_{\text{ISM}}^{\text{style}}(\theta) = \mathbb{E}_{t,c}\left[\omega(t)\delta(x_t, x_{t-1}; y, I_{ref}, y_{ref}, t, t-1)\frac{\partial g(\theta, c)}{\partial \theta}\right]. \tag{6}$$

The gradient updating direction $\delta(x_t, x_{t-1}; y, I_{ref}, y_{ref}, t, t-1)$ incorporates the style and content from the reference image I_{ref} as well as two text prompts y and y_{ref}. It is formulated as:

$$\begin{aligned}&\delta(x_t, x_{t-1}; y, I_{ref}, y_{ref}, t, t-1) \\ &= \epsilon_\phi(x_t; t) - \epsilon_\phi(x_{t-1}; t-1) \\ &+ \lambda_{cfg}(\epsilon_\phi(x_t; t, y) - \epsilon_\phi(x_t; t, y_{ref})) \\ &+ \delta_{style}(x_t; y, I_{ref}, y_{ref}, t),\end{aligned} \tag{7}$$

where y_{ref} indicates the unintended content information. We integrate y_{ref} into the CFG term $\epsilon_\phi(x_t; t, y) - \epsilon_\phi(x_t; t, y_{ref})$ as a negative prompt to reduce the content leakage artifacts. Besides, we explicitly employ a novel style guidance $\delta_{style}(x_t; y, I_{ref}, y_{ref}, t)$ to direct the style of the rendered image to the desired one. The style guidance aims to reduce the score distribution divergence between the rendered images and the images with the desired style. Inspired by the classifier guidance, our style guidance can be formulated as:

$$\begin{aligned}&\delta_{style}(x_t; y, I_{ref}, y_{ref}, t) \\ &= \lambda_{style}(\epsilon_{style}(x_t; t, y, I_{ref}, y_{ref}) - \epsilon_\phi(x_t; t)),\end{aligned} \tag{8}$$

where λ_{style} is a weight factor, and $\epsilon_{style}(x_t; t, y, I_{ref}, y_{ref})$ predicts the distribution of the required style images.

3.3 Style Score Distribution

To achieve the style distribution for ϵ_{style}, a possible way is to train a style-conditioned diffusion model, but it is time-consuming. Instead,

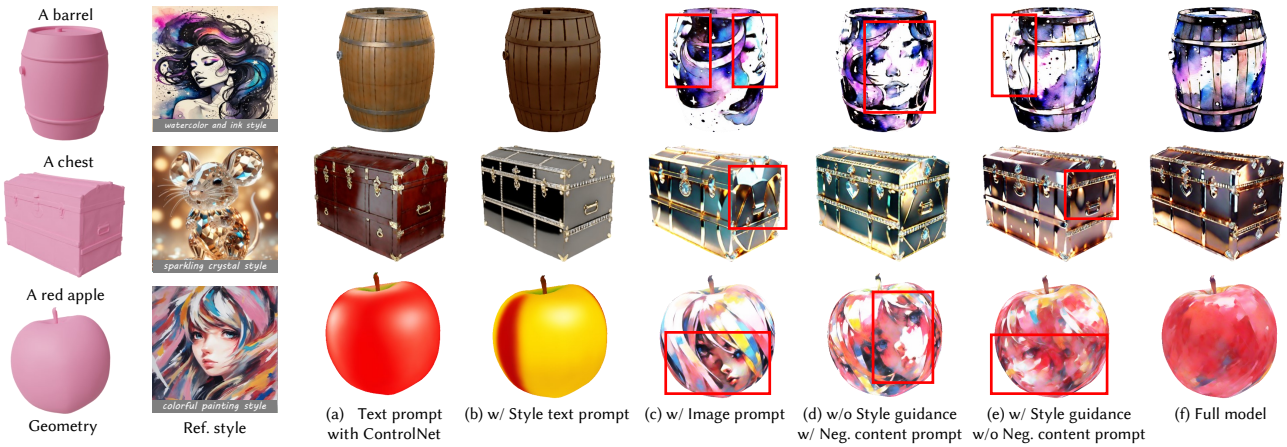


Fig. 3. **Ablation study on style guidance**. (a) Baseline for text-to-texture. (b) Use "in xxx style" text prompts for style guidance. (c) Add the whole image prompt as guidance. (d) Add our style guidance strategy. (e) Add content embedding of the reference image as a negative prompt. (f) Full model with the style guidance strategy and content embedding of the reference image as a negative prompt.

inspired by [Wang et al. 2024; Ye et al. 2023], we shift the original non-style distribution of a pre-trained diffusion model to the desired one by injecting information from the reference image into the diffusion model. Therefore, the core requirement is to extract style information from the reference image while disregarding content information.

Existing 2D style image generation studies [Wang et al. 2024; Ye et al. 2023] have explored that the cross-attention mechanism in different transformer layers of a diffusion model exerts different effects on the content and style. Therefore, the stylized result can be achieved by injecting the features of the reference image into the layers that are responsible for style effects. However, a transformer layer can be in charge of both style and content because of the ambiguity in them. Leveraging such a layer to inject the reference image feature may introduce unintended content, while ignoring it may result in inaccuracies in style expressiveness, such as color tone shifting. To address this, we aim to incorporate as many layers that are responsible for style effects as possible to maintain style expressiveness. The appendix contains detailed information about our leveraged transformer layers. Simultaneously, to mitigate the influence of content from adding these layers, we propose explicitly disentangling the style and content from the image feature to extract a cleaner style.

To disentangle the content and style, we leverage the text content prompt y_{ref} as the content guidance. Specifically, based on the multi-modal applications of the CLIP space, we encode the reference image and the text content prompt into the same space using a CLIP image encoder and a CLIP text encoder, respectively, resulting in image embedding and content embedding. While the content embedding encodes the majority of the content information of the image, text-based descriptions cannot align accurately with the abundant image information. Therefore, simply driving the image embedding towards the opposite of the content embedding direction cannot eliminate the content correctly. Driving too little does not influence the image content, while driving too much may alter the reference image's color tone. To this end, we propose to decompose the image embedding into two components, with one component aligning with the content embedding explicitly. Specifically, we employ an orthogonal decomposition for content removal (ODCR):

$$f_g^{ref} = E_{CLIP}^{img}(I_{ref}), \quad f_c^{ref} = E_{CLIP}^{text}(y_{ref}),$$
$$f_s^{ref} = f_g^{ref} - \frac{f_c^{ref}(f_g^{ref})^T f_c^{ref}}{||f_c^{ref}||_2^2}, \tag{9}$$

where f_g^{ref} is the reference image embedding extracted by the CLIP's image encoder E_{CLIP}^{img}, and f_c^{ref} is the content embedding extracted by the CLIP's text encoder E_{CLIP}^{text}. After ODCR, we remain only the f_s^{ref} to guide the diffusion model. The experiments in Sec. 4.1.1 demonstrate the superiority of our decomposition.

4 EXPERIMENTS AND RESULTS

4.1 Ablation Study

We first conduct an ablation study to show the style effectiveness of each component of our method, including using y_{ref} as the negative prompt and using our style guidance δ_{style} for disentangling and injecting the style. Then we dive into our style guidance to validate the effectiveness of our chosen transformer layers and the image embedding decomposition. Next, we validate that the geometry-aware ControlNet is beneficial to 3D consistency. Lastly, we conduct an experiment to show that using ISM achieves higher-quality results.

4.1.1 Style effectiveness for each component. As a baseline, we use a non-style text-to-texture generation that uses an ISM-based framework with a geometry-aware ControlNet to produce three outcomes, with "a red apple", "a chest", and "a barrel" as the textual conditions, respectively. The results shown in Fig. 3 (a) present multi-view consistency while not presenting any specific style. Then in Fig. 3 (b), we add textual descriptions of the desired style in the prompt and hence these prompts become "a barrel in watercolor and ink style", "a chest in sparkling crystal style", and "a red apple in a colorful painting style". Although the results in Fig. 3 (b) demonstrate some

Fig. 4. Stylized texture results obtained using various transformer layer style injection strategies. The Prompts are "a cupcake in ice and snow covered style" and "a wooden treasure chest with metal accents and locks in colorful drawing style".

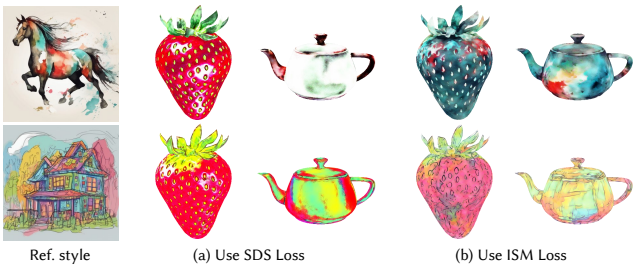


Fig. 5. Results using our style-content decoupling method with SDS loss (a) and ISM loss (b) for the prompts "a strawberry/teapot in colorful graffiti style" and "a strawberry/teapot in Chinese ink paint style".

color changes compared to the baseline, they fail to convey the style effectively. Image-based texture generation methods, such as [Ye et al. 2023], take the reference image as the input and can achieve vivid style. However, as shown in Fig. 3 (c), without disentangling the content and style, the content information of the reference image is incorrectly retained in the results. We then showcase two variants of our method, one removes our style guidance (Fig. 3 (d)) and another removes the negative prompt of the CFG term (Fig. 3 (e)). Both methods achieve a vivid style and alleviate the content leakage artifacts. Lastly, our method shown in Fig. 3 (f) exhibits high-quality results with vivid style while not presenting artifacts.

4.1.2 Style guidance. Our style guidance δ_{style} is carefully designed to preserve style expressiveness while not leading to content leakage by two aspects. Firstly, in terms of style injection in transformer layers, unlike existing 2D style image generation methods that do not consider transformer layers that are more responsible for content than style, we use all transformer layers that impact style to achieve style consistency in multiple views. As shown in Fig. 4, incorporating only the transformer layers used by 2D style image generation methods can result in a color tone that deviates from that of the reference image. Secondly, we explicitly decompose the reference image embedding within the CLIP space to disentangle the style and content. As shown in Fig. 6 (a), incorporating the complete image embedding into the diffusion model leads to severe

Fig. 6. Stylized texture results achieved using different content removal strategies in CLIP space. The prompts are "a hand bag in watercolor sketch style" and "a pot in a colorful painting style".

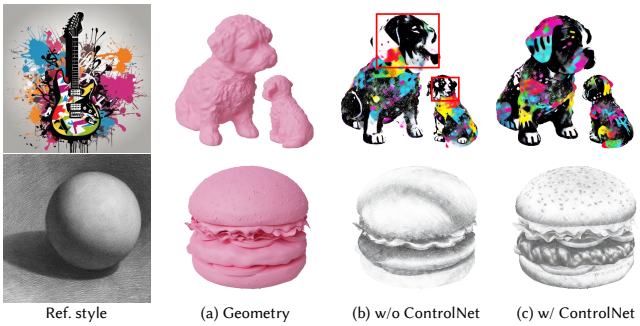


Fig. 7. **Ablation study on geometric ControlNet.** The prompts are "a dog in graffiti style" and "a hamburger in sketch style".

content leakage artifacts. Besides, without disentangling the style and content, driving the image embedding by the content embedding easily results in artifacts. For instance, greatly altering the image embedding can lead to inaccurate color expressiveness (Fig. 6 (b)), while slight modifications cause content leakage (Fig. 6 (c)). In Fig. 6 (d), our method presents a superior performance in both style expressiveness and content removal.

4.1.3 ISM vs SDS. To achieve superior quality, we utilize an ISM-based optimization framework instead of SDS [Poole et al. 2022]. As illustrated in Fig. 5, replacing our ISM loss with the SDS loss exhibits over-saturation and over-smoothness and severely undermines the style expressiveness.

4.1.4 Geometry-aware ControlNet. Our method uses a geometry-aware ControlNet that receives the rendered depth and normal map as inputs. To validate its effectiveness in preserving 3D consistency and geometrical details, we conduct an experiment using a vanilla diffusion model. As shown in Fig. 7, the geometry-aware ControlNet greatly enhances the detail of textures, particularly in models with complex geometries (e.g., the hamburger). Furthermore, it also aids in eliminating the Janus problem as shown in the first row of Fig. 7.

4.2 Comparison

We compare our method to several state-of-the-art methods for image-guided 3D generation, namely TEXTure [Richardson et al. 2023], TextureDreamer [Yeh et al. 2024], IPDreamer [Zeng et al. 2023b], and a text-based texture generation method SyncMVD [Liu et al. 2023a]. Since IPDreamer [Zeng et al. 2023b] synthesizes 3D

Fig. 8. Qualitative comparison to TEXTure [Richardson et al. 2023], TextureDreamer [Yeh et al. 2024], IPDreamer [Zeng et al. 2023b], and SyncMVD [Liu et al. 2023a].

Table 1. **User study results.** Participants are asked to evaluate the overall quality, style fidelity, and content removal of the generated results by giving scores ($\in [1, 5]$) to the rendering videos. This table shows the average scores given by 37 participants.

Method	Overall Quality↑	Style Fidelity↑	Content Removal↑
TEXTure	2.62	2.01	2.83
TextureDreamer	2.29	1.93	2.68
IPDreamer	3.02	3.45	3.01
SyncMVD	3.07	2.67	3.23
Ours	**4.60**	**4.61**	**4.36**

Table 2. **Quantitative comparison results.** We utilize the Gram Matrix Distance to measure style fidelity, and use the CLIP score to measure the semantic alignment between the prompts and the results.

Method	Gram Matrix Distance↓	CLIP Score↑
TEXTure	0.830	68.01
TextureDreamer	0.947	68.57
IPDreamer	0.910	61.81
SyncMVD	0.920	69.60
Ours	**0.723**	**73.66**

geometry in addition to texture, we fix the geometry and concentrate solely on texture synthesis. Besides, SyncMVD [Liu et al. 2023a] synthesizes 2D images across multiple views rather than using score distillation sampling and hence cannot incorporate our style guidance. For a fair comparison, we incorporate a 2D image-guided generation method [Wang et al. 2024] into SyncMVD.

4.2.1 Qualitative Comparison. Fig. 8 and Fig. 10 provide qualitative comparisons between the baseline methods and our proposed approach. Both TEXTure and TextureDreamer utilize reference images to fine-tune the diffusion model, with the generated texture heavily relying on the performance of fine-tuning. However, given only a single reference image, the fine-tuned diffusion model either overfits or fails to accurately extract the image style, leading to incorrect results when applied to a mesh whose subject does not match the reference image. IPDreamer does not separate the style and content of the reference image during generation, resulting in a significant content leakage issue. Additionally, the usage of the SDS leads to over-saturation. While SyncMVD can synthesize multi-view images that exhibit some extent of the style, it suffers from balancing between the multi-view consistency, the image guidance and the classifier term, leading to overly smooth, detail-lacking, and style drifting results. In contrast, our results demonstrate superior performance in terms of detail representation and style fidelity compared to all other methods.

4.2.2 Quantitative Comparison. We first conduct user study using 12 styles and 24 meshes to evaluate the results of all methods regarding quality, style fidelity, and content removal. For each style, we use each method to generate textures for 2 meshes, respectively. We ask 37 participants to assign a score range from 1 to 5 to the synthesized results of all methods. The higher score indicates the better performance. The results are shown in Tab. 1. Among all methods, our method achieves the superior performance in terms of all metrics.

In addition to the user study, we use the common metrics for image generation methods to evaluate all methods in terms of style fidelity and semantic alignment. For 25 randomly chosen styles, we use each style to generate stylized textures for 4 unique, randomly selected meshes from Objaverse [Deitke et al. 2023], totaling 100 different results. We then render four views per result to compute the metrics. The style's fidelity is measured by the Gram metrics difference [Johnson et al. 2016] between the rendered images and

the reference images. Besides, the semantic alignment between the prompts and the rendered image is measured by the CLIP Score [Hessel et al. 2021]. As shown in Tab. 2, our method outperforms all other methods in achieving the best style fidelity and text alignment. The details of these metrics are outlined in the supplement material.

4.3 Results

An NVIDIA RTX 4090 GPU is used for the optimization process, which takes about 15 minutes to synthesize a texture map for each mesh. We demonstrate the robustness of our method using a diverse range of reference images, including various artistic styles such as "sketching" and "ink wash painting", different materials like "gold" and "wool", as well as various patterns and brush strokes. The generated results shown in Fig. 11 maintain multi-view consistency, align with the geometric details of the models, and adhere to the style of the reference image.

In addition, we demonstrate that our method can be practically used for games or films, which requires generating consistent style for all meshes. As shown in Fig. 1, we create textures for various objects that share the same style given a reference image, resulting in scenes that are harmonious and aesthetically pleasing.

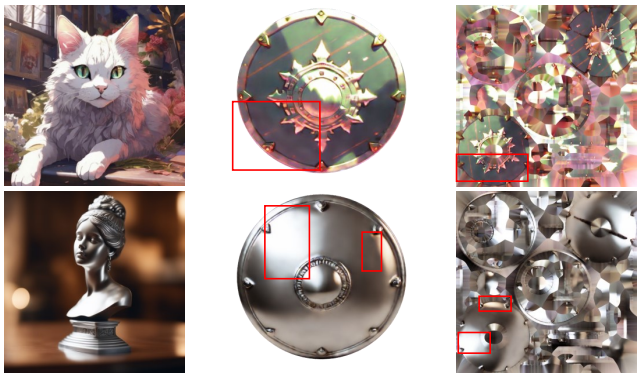
5 LIMITATIONS AND CONCLUSIONS

5.1 Limitations

Despite the successful generation of high-quality textures that align with the style of the reference image, our method presents several limitations. To begin, unlike PBR materials generation methods [Zhang et al. 2024c,b], the influence of style prevents us from identifying a universally applicable rendering model, making it difficult to define and decouple the highlights and shadows contained in textures. This issue may result in baked-in highlights or shadows in the generated textures, as shown in Fig. 9. Second, our method's distillation time is relatively long, which limits its use in an interactive environment. Future work could potentially accelerate our method by integrating recent advancements in diffusion models and representations. Finally, as style is the result of a combination of various elements (including material, brush strokes, tone, and painting style), our method is unable to extract or adjust any of these elements individually.

5.2 Conclusions

This paper presents StyleTex, a novel stylized texture generation approach for the given mesh, guided by a single reference image and text prompts. StyleTex leverages an ISM-based generative framework, incorporating both style guidance and geometric control. The key advantage of our method is a novel strategy for disentangling style and content information, which effectively addresses the prevalent issues of content leakage and style drift in 3D stylized textures. By utilizing a single stylized image as the reference, StyleTex can generate textures that exhibit similar styles, thereby enabling the automatic creation of visually compelling and immersive virtual environments for games or films.



Ref. style (a) Textured mesh (b) Texture map

Fig. 9. Artifacts caused by baked-in highlights or shadows in generated textures (b). The red boxes represent unintended baked-in shadows (a,b) (upper) and highlights (a,b) (bottom).

ACKNOWLEDGMENTS

Xiaogang Jin was supported by Key R&D Program of Zhejiang (No. 2024C01069).

REFERENCES

Aishwarya Agarwal, Srikrishna Karanam, Tripti Shukla, and Balaji Vasan Srinivasan. 2023. An Image Is Worth Multiple Words: Multi-Attribute Inversion for Constrained Text-to-Image Synthesis. *arXiv preprint arXiv:2311.11919* (2023).

Jie An, Siyu Huang, Yibing Song, Dejing Dou, Wei Liu, and Jiebo Luo. 2021. ArtFlow: Unbiased image style transfer via reversible neural flows. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2021*. IEEE, 862–871.

Tianshi Cao, Karsten Kreis, Sanja Fidler, Nicholas Sharp, and KangXue Yin. 2023. TexFusion: Synthesizing 3D Textures with Text-Guided Image Diffusion Models. In *2023 IEEE/CVF International Conference on Computer Vision, ICCV 2023*. IEEE, 4146–4158.

Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. 2017. StyleBank: An Explicit Representation for Neural Image Style Transfer. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*. 2770–2779.

Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. 2023c. Text2Tex: Text-driven Texture Synthesis via Diffusion Models. In *2023 IEEE/CVF International Conference on Computer Vision, ICCV 2023*. IEEE, 18512–18522.

Jingwen Chen, Yingwei Pan, Ting Yao, and Tao Mei. 2023b. ControlStyle: Text-Driven Stylized Image Generation Using Diffusion Priors. *arXiv preprint arXiv:2311.05463* (2023).

Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. 2023a. Fantasia3D: Disentangling Geometry and Appearance for High-quality Text-to-3D Content Creation. In *2023 IEEE/CVF International Conference on Computer Vision, ICCV 2023*. IEEE, 22189–22199.

Tian Qi Chen and Mark W. Schmidt. 2016. Fast Patch-based Style Transfer of Arbitrary Style. *arXiv preprint arXiv:1612.04337* (2016).

Zhiqin Chen, Kangxue Yin, and Sanja Fidler. 2022. AUV-Net: Learning Aligned UV Maps for Texture Transfer and Synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022*. IEEE, 1455–1464.

Evgeniia Cheskidova, Aleksandr Arganaidi, Daniel-Ionut Rancea, and Olaf Haag. 2023. Geometry Aware Texturing. In *SIGGRAPH Asia 2023 Posters (SA '23)*. Association for Computing Machinery, Article 21, 2 pages. https://doi.org/10.1145/3610542.3626152

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. 2023. Objaverse: A Universe of Annotated 3D Objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*. IEEE, 13142–13153.

Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. 2017. A Learned Representation For Artistic Style. In *International Conference on Learning Representations*.

Yarden Frenkel, Yael Vinker, Ariel Shamir, and Daniel Cohen-Or. 2024. Implicit Style-Content Separation using B-LoRA. *arXiv preprint arXiv:2403.14572* (2024).

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. 2023. An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion. In *International Conference on Learning Representations*.

Chenjian Gao, Boyan Jiang, Xinghui Li, Yingpeng Zhang, and Qian Yu. 2024. GenesisTex: Adapting Image Denoising Diffusion to Texture Space. In *Proceedings of the IEEE/CVF*

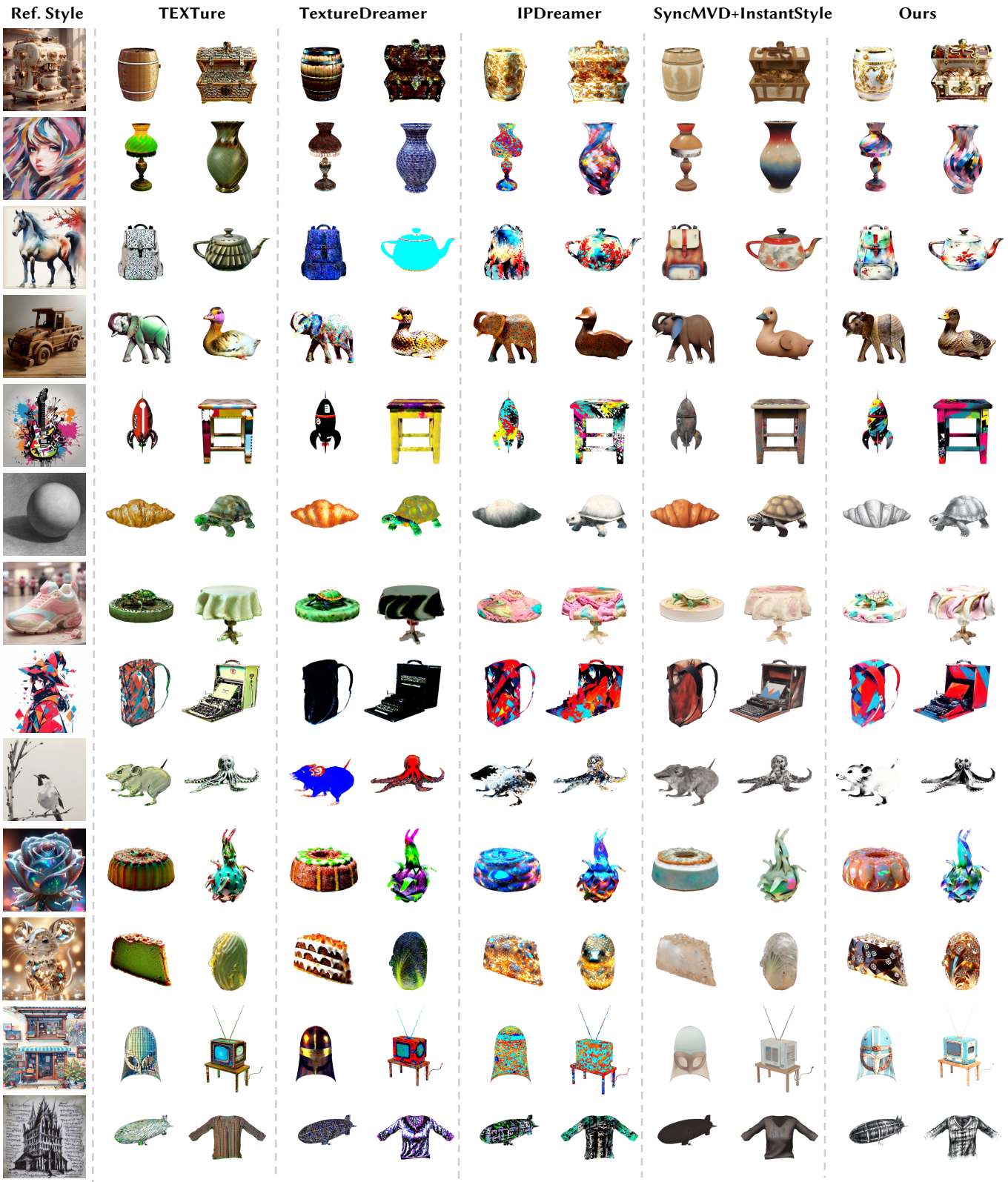


Fig. 10. More qualitative comparison with TEXTure [Richardson et al. 2023], TextureDreamer [Yeh et al. 2024], IPDreamer [Zeng et al. 2023b], and SyncMVD [Liu et al. 2023a].

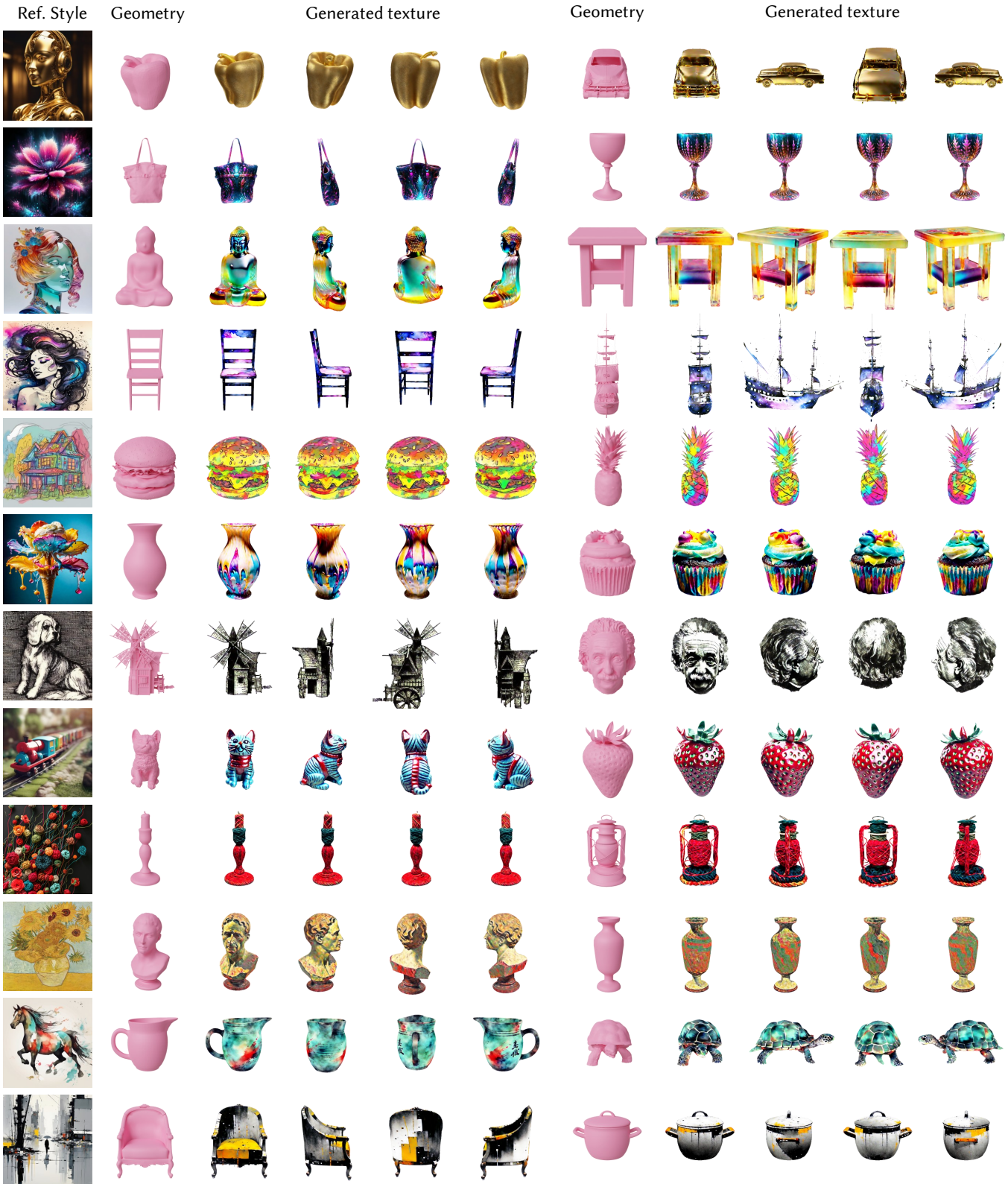


Fig. 11. Results of StyleTex. For each style, we generate textures for two meshes and showcase four different rendered views.

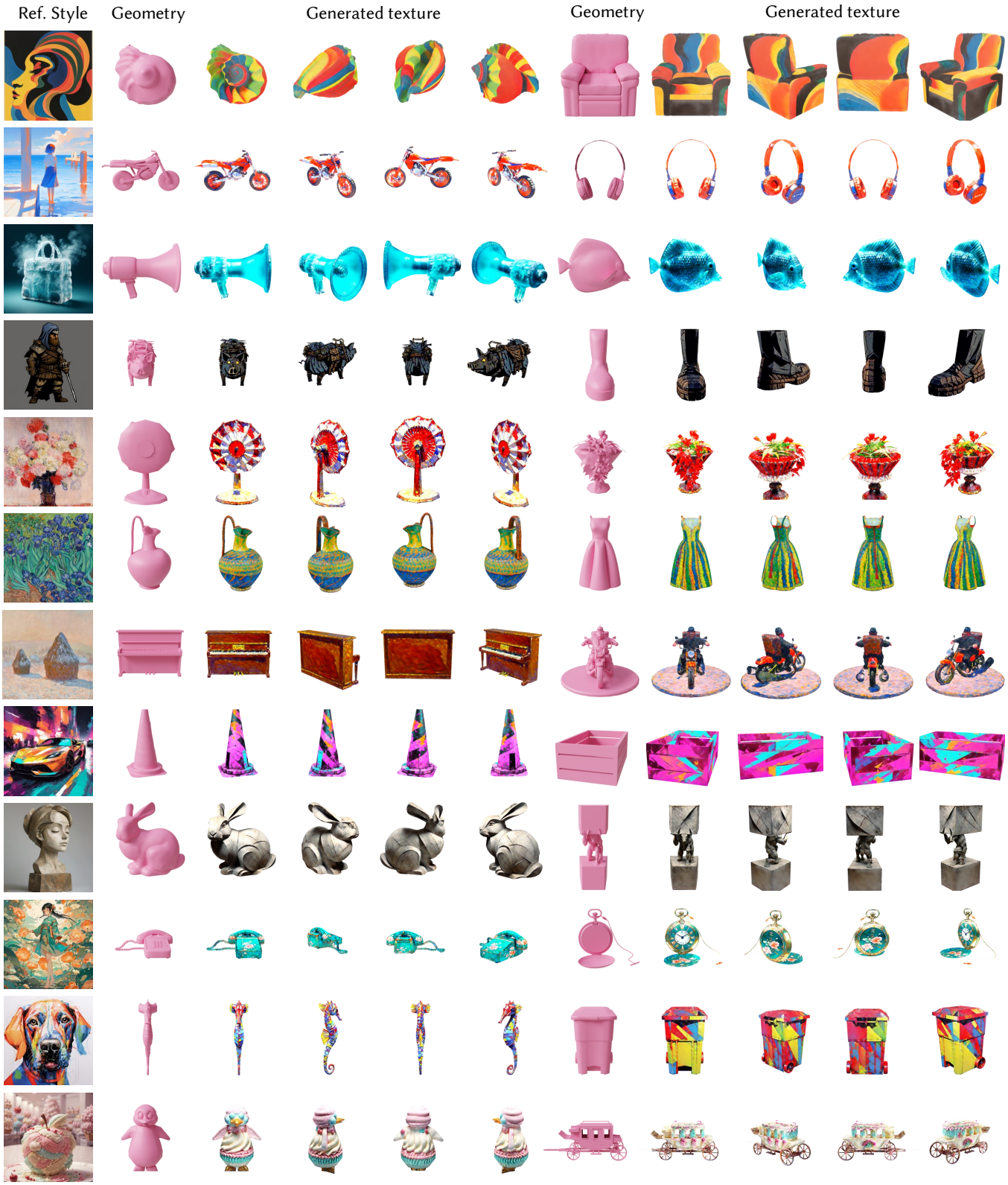


Fig. 12. Results of StyleTex. For each style, we generate textures for two meshes and showcase four different rendered views.

Conference on Computer Vision and Pattern Recognition, CVPR 2024. 4620–4629.

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. 2016. Image Style Transfer Using Convolutional Neural Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*. 2414–2423.

Shuyang Gu, Congliang Chen, Jing Liao, and Lu Yuan. 2018. Arbitrary Style Transfer with Deep Feature Reshuffle. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2018*. 8222–8231.

Yanhui Guo, Xinxin Zuo, Peng Dai, Juwei Lu, Xiaolin Wu, Youliang Yan, Songcen Xu, Xiaofei Wu, et al. 2023. Decorate3D: Text-Driven High-Quality Texture Generation for Mesh Decoration in the Wild. In *Advances in Neural Information Processing Systems, NeurIPS 2023*, Vol. 36. 36664–36676.

Feihong He, Gang Li, Mengyuan Zhang, Leilei Yan, Lingyu Si, and Fanzhang Li. 2024. Freestyle: Free lunch for text-guided style transfer using diffusion models. *arXiv preprint arXiv:2401.15636* (2024).

Amir Hertz, Andrey Voynov, Shlomi Fruchter, and Daniel Cohen-Or. 2024. Style Aligned Image Generation via Shared Attention. *arXiv preprint arXiv:2312.02133* (2024).

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: A Reference-free Evaluation Metric for Image Captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021*. 7514–7528.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems, NeurIPS 2020*, Vol. 33. 6840–6851.

Jonathan Ho and Tim Salimans. 2021. Classifier-Free Diffusion Guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*.

Lukas Höllein, Justin Johnson, and Matthias Nießner. 2022. StyleMesh: Style Transfer for Indoor 3D Scene Reconstructions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022*. IEEE, 6188–6198.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Hsin-Ping Huang, Hung-Yu Tseng, Saurabh Saini, Maneesh Singh, and Ming-Hsuan Yang. 2021. Learning to stylize novel views. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021*. IEEE, 13849–13858.

Xun Huang and Serge Belongie. 2017. Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization. In *2017 IEEE International Conference on Computer Vision, ICCV 2017*. IEEE, 1510–1519.

Yi-Hua Huang, Yue He, Yu-Jie Yuan, Yu-Kun Lai, and Lin Gao. 2022. StylizedNeRF: Consistent 3D Scene Stylization as Stylized NeRF via 2D-3D Mutual Learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022*. 18321–18331.

Jaeseok Jeong, Junho Kim, Yunjey Choi, Gayoung Lee, and Youngjung Uh. 2024. Visual Style Prompting with Swapping Self-Attention. *arXiv preprint arXiv:2402.12974* (2024).

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *Computer Vision - ECCV 2016 - 14th European Conference (Lecture Notes in Computer Science, Vol. 9906)*. 694–711.

Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Neural 3D Mesh Renderer. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018*. Computer Vision Foundation / IEEE Computer Society, 3907–3916.

Nicholas Kolkin, Michal Kucera, Sylvain Paris, Daniel Sykora, Eli Shechtman, and Greg Shakhnarovich. 2022. Neural Neighbor Style Transfer. *arXiv preprint arXiv:2203.13215* (2022).

adn Sanakoyeu Artsiom Kotovenko, Dmytro, Sabine Lang, and Björn Ommer. 2019. Content and Style Disentanglement for Artistic Style Transfer. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019*. IEEE, 4421–4430.

Cindy Le, Congrui Hetang, Ang Cao, and Yihui He. 2023. EucliDreamer: Fast and High-Quality Texturing for 3D Models with Stable Diffusion Depth. *arXiv preprint arXiv:2311.15573* (2023).

Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. 2017. Universal Style Transfer via Feature Transforms. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*. 385–395.

Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. 2024. LucidDreamer: Towards High-Fidelity Text-to-3D Generation via Interval Score Matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024*. 6517–6526.

Kunhao Liu, Fangneng Zhan, Yiwen Chen, Jiahui Zhang, Yingchen Yu, Abdulmotaleb El Saddik, Shijian Lu, and Eric P Xing. 2023b. StyleRF: Zero-Shot 3D Style Transfer of Neural Radiance Fields. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*. 8338–8348.

Yuxin Liu, Minshan Xie, Hanyuan Liu, and Tien-Tsin Wong. 2023a. Text-Guided Texturing by Synchronized Multi-View Diffusion. *arXiv preprint arXiv:2311.12891* (2023).

Yufei Liu, Junwei Zhu, Junshu Tang, Shijie Zhang, Jiangning Zhang, Weijian Cao, Chengjie Wang, Yunsheng Wu, and Dongjin Huang. 2024. TexDreamer: Towards Zero-Shot High-Fidelity 3D Human Texture Generation. *arXiv preprint arXiv:2403.12906* (2024).

Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. 2023. Latent-NeRF for Shape-Guided Generation of 3D Shapes and Textures. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*. 12663–12673.

Fangzhou Mu, Jian Wang, Yicheng Wu, and Yin Li. 2022. 3D Photo Stylization: Learning to Generate Stylized Novel Views from a Single Image. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022*. 16252–16261.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)* 41, 4, Article 102 (2022), 15 pages.

Thu Nguyen-Phuoc, Feng Liu, and Lei Xiao. 2022. SNeRF: Stylized Neural Implicit Representations for 3D Scenes. *arXiv preprint arXiv:2207.02363* (2022).

Dae Young Park and Kwang Hee Lee. 2019. Arbitrary Style Transfer With Style-Attentional Networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2019*. 5873–5881.

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

Tianhao Qi, Shancheng Fang, Yanze Wu, Hongtao Xie, Jiawei Liu, Lang Chen, Qian He, and Yongdong Zhang. 2024. DEADiff: An Efficient Stylization Diffusion Model with Disentangled Representations. *arXiv preprint arXiv:2403.06951* (2024).

Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. 2023. Texture: Text-guided texturing of 3d shapes. In *ACM SIGGRAPH 2023 Conference Proceedings* (Los Angeles, CA, USA) *(SIGGRAPH '23)*. Association for Computing Machinery, New York, NY, USA, Article 54, 11 pages. https://doi.org/10.1145/3588432.3591503

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-Resolution Image Synthesis with Latent Diffusion Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022*. IEEE, 10674–10685.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. 2023. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*. 22500–22510.

Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. 2023. ZipLoRA: Any Subject in Any Style by Effectively Merging LoRAs. *arXiv preprint arxiv:2311.13600* (2023).

Yawar Siddiqui, Justus Thies, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. 2022. Texturify: Generating Textures on 3D Shape Surfaces. In *Computer Vision – ECCV 2022: 17th European Conference* (Tel Aviv, Israel). 72–88.

Kihyuk Sohn, Nataniel Ruiz, Kimin Lee, Daniel Castro Chin, Irina Blok, Huiwen Chang, Jarred Barber, Lu Jiang, Glenn Entis, Yuanzhen Li, et al. 2024. StyleDrop: Text-to-Image Generation in Any Style. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*. Article 2920, 30 pages.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising Diffusion Implicit Models. In *International Conference on Learning Representations*.

Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor S. Lempitsky. 2016. Texture Networks: Feed-forward Synthesis of Textures and Stylized Images. *arXiv preprint arXiv:1603.03417* (2016).

Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. 2023. $P+$: Extended Textual Conditioning in Text-to-Image Generation. *arXiv preprint arXiv:2303.09522* (2023).

Haofan Wang, Qixun Wang, Xu Bai, Zekui Qin, and Anthony Chen. 2024. InstantStyle: Free Lunch towards Style-Preserving in Text-to-Image Generation. *arXiv preprint arXiv:2404.02733* (2024).

Zhouxia Wang, Xintao Wang, Liangbin Xie, Zhongang Qi, Ying Shan, Wenping Wang, and Ping Luo. 2023. StyleAdapter: A Single-Pass LoRA-Free Model for Stylized Image Generation. *arXiv preprint arxiv:2309.01770* (2023).

Jinbo Wu, Xing Liu, Chenming Wu, Xiaobo Gao, Jialun Liu, Xinqi Liu, Chen Zhao, Haocheng Feng, Errui Ding, and Jingdong Wang. 2024. TexRO: Generating Delicate Textures of 3D Models by Recursive Optimization. *arXiv preprint arXiv:2403.15009* (2024).

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. 2023. IP-Adapter: Text Compatible Image Prompt Adapter for Text-to-Image Diffusion Models. *arXiv preprint arxiv:2308.06721* (2023).

Yu-Ying Yeh, Jia-Bin Huang, Changil Kim, Lei Xiao, Thu Nguyen-Phuoc, Numair Khan, Cheng Zhang, Manmohan Chandraker, Carl S Marshall, Zhao Dong, et al. 2024. TextureDreamer: Image-guided Texture Synthesis through Geometry-aware Diffusion. *arXiv preprint arXiv:2401.09416* (2024).

Kangxue Yin, Jun Gao, Maria Shugrina, Sameh Khamis, and Sanja Fidler. 2021. 3DStyleNet: Creating 3D Shapes with Geometric and Texture Style Variations. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021*. IEEE, 12436–12445.

Kim Youwang, Tae-Hyun Oh, and Gerard Pons-Moll. 2023. Paint-it: Text-to-Texture Synthesis via Deep Convolutional Texture Map Optimization and Physically-Based Rendering. *arXiv preprint arXiv:2312.11360* (2023).

Bohan Zeng, Shanglin Li, Yutang Feng, Hong Li, Sicheng Gao, Jiaming Liu, Huaxia Li, Xu Tang, Jianzhuang Liu, and Baochang Zhang. 2023b. Ipdreamer: Appearance-controllable 3d object generation with image prompts. *arXiv preprint arXiv:2310.05375* (2023).
Xianfang Zeng, Xin Chen, Zhongqi Qi, Wen Liu, Zibo Zhao, Zhibin Wang, Bin Fu, Yong Liu, and Gang Yu. 2023a. Paint3D: Paint Anything 3D with Lighting-Less Texture Diffusion Models. *arXiv preprint arXiv:2312.13913* (2023).
Dingxi Zhang, Zhuoxun Chen, Yujian Yuan, Fang-Lue Zhang, Zhenliang He, Shiguang Shan, and Lin Gao. 2024a. StylizedGS: Controllable Stylization for 3D Gaussian Splatting. *arXiv preprint arXiv:2404.05220* (2024).
Hang Zhang and Kristin Dana. 2019. Multi-Style Generative Network for Real-Time Transfer. In *Computer Vision – ECCV 2018 Workshops: Munich*. 349–365.
Kai Zhang, Nick Kolkin, Sai Bi, Fujun Luan, Zexiang Xu, Eli Shechtman, and Noah Snavely. 2022. ARF: Artistic Radiance Fields. In *Computer Vision – ECCV 2022: 17th European Conference*. 717–733.
Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. 2023b. Adding Conditional Control to Text-to-Image Diffusion Models. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023*. IEEE, 3813–3824.
Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. 2024c. CLAY: A Controllable Large-scale Generative Model for Creating High-quality 3D Assets. *ACM Trans. Graph.* 43, 4, Article 120 (2024), 20 pages. https://doi.org/10.1145/3658146
Yuxin Zhang, Nisha Huang, Fan Tang, Haibin Huang, Chongyang Ma, Weiming Dong, and Changsheng Xu. 2023a. Inversion-based Style Transfer with Diffusion Models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*. IEEE, 10146–10156.
Yuqing Zhang, Yuan Liu, Zhiyu Xie, Lei Yang, Zhongyuan Liu, Mengzhou Yang, Runze Zhang, Qilong Kou, Cheng Lin, Wenping Wang, and Xiaogang Jin. 2024b. DreamMat: High-quality PBR Material Generation with Geometry- and Light-aware Diffusion Models. *ACM Trans. Graph.* 43, 4, Article 39 (2024), 18 pages. https://doi.org/10.1145/3658170

A APPENDIX

A.1 Detailed Difference with InstantStyle

Fig. 13. Results using different types of layers in InstantStyle.

InstantStyle [Wang et al. 2024] categorizes attention layers that influence style into two types: style-only and spatial layout. In 2D image generation, using full layers can introduce both the reference image's content and style information (see Fig. 13 (a)). Employing both the style-only and layout layers may introduce stylistic information as well as spatial structural information (see Fig. 13 (b)), whereas only using the style-only layer may result in minor tonal discrepancies (see Fig. 13 (c)). In 3D contexts, excessive structural information from layout layers may result in content leakage, and the absence of tonal information from style-only layers can cause severe tonal shifts. Furthermore, InstantStyle uses a simple feature subtraction technique to separate style and content. The style feature is obtained by subtracting the text embedding from the image embedding, resulting in partial content information leakage.

Unlike their approach, we use InstantStyle's style-only and layout layers, as well as additional layers [Agarwal et al. 2023; Voynov et al. 2023], to preserve complete style information and avoid tonal shifts. To remove as much structural and content information from the reference image as possible, we use ODCR to extract style features. Furthermore, the content description of the reference image serves as a negative prompt during the distillation process.

A.2 Additional Transformer Layers

The cross-attention layers Instant Style uses for style injection including:

- down_blocks.2
- mid_block.attention.0
- up_block.1

In StyleTex, we expand the number of cross-attention layers used for style injection, including:

- down_blocks.1.attentions.0
- All layers in up_block

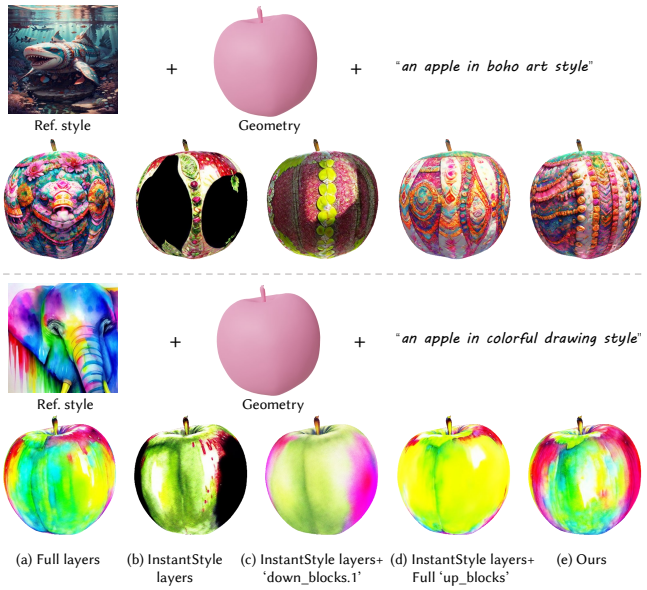


Fig. 14. The impact of the additional transformer layers leveraged in our method.

To evaluate the impact of the additional transformer layers used, we conducted an experiment in which we modified the transformer layers in our full model. The results are presented in Fig. 14. Fig. 14 (a) demonstrates that injecting style information into all layers results in content leakage issues. Fig. 14 (b) shows that using only the original injection layer of Instant Style leads to style drift and black areas due to the removal of too many layers in the style injection. By solely adding "down_blocks.1.attentions.0" or "up_blocks", as depicted in Fig. 14 (c) and (d), respectively, the black area is effectively removed; however, a slight color shift still occurs. In contrast, using the additional layers as we did in our proposed approach produces results that more closely align with the reference image while avoiding content leakage.