International Journal of Computer Vision (2025) 133:5026–5061
https://doi.org/10.1007/s11263-025-02403-9

Advances in 3D Neural Stylization: A Survey

Yingshu Chen[1] · Guocheng Shao[1] · Ka Chun Shum[1] · Binh-Son Hua[2] · Sai-Kit Yeung[1]

Received: 30 March 2024 / Accepted: 20 February 2025 / Published online: 28 March 2025

Abstract
Modern artificial intelligence offers a novel and transformative approach to creating digital art across diverse styles and modalities like images, videos and 3D data, unleashing the power of creativity and revolutionizing the way that we perceive and interact with visual content. This paper reports on recent advances in stylized 3D asset creation and manipulation with the expressive power of neural networks. We establish a taxonomy for neural stylization, considering crucial design choices such as scene representation, guidance data, optimization strategies, and output styles. Building on such taxonomy, our survey first revisits the background of neural stylization on 2D images, and then presents in-depth discussions on recent neural stylization methods for 3D data, accompanied by a benchmark evaluating selected mesh and neural field stylization methods. Based on the insights gained from the survey, we highlight the practical significance, open challenges, future research, and potential impacts of neural stylization, which facilitates researchers and practitioners to navigate the rapidly evolving landscape of 3D content creation using modern artificial intelligence.

Keywords 3D stylization · Neural style transfer · Neural stylization

1 Introduction

Digital art and visual design have been prevailing in our daily living spaces, expressing visually captivating aesthetics, unique tastes, and emotions of human beings. With the prosperity of artificial intelligence (AI), there emerges a new generation of toolsets for visual content creation such as generative image synthesis (Stable Diffusion, DALL·E 3, Midjourney) (Rombach et al., 2022; OpenAI, 2023; Midjourney, 2023), and video synthesis (Sora, RunwayML) (OpenAI, 2024; Runway, 2023). AI-based visual content creation can also be extended to the 3D domain, notably by lifting images to 3D scenes (LUMA AI) (Luma, 2023), and creative text-guided 3D generation and design (DreamFusion, Meshy, SplineAI) (Poole et al., 2023; Meshy, 2024; Spline, 2023).

It's noteworthy that the nature of visual concepts in our living space is tied to a critical factor: style. Formally, style is a way to express individuality and creativity in different mediums and practices. In relevant industries like animation, architectural and interior design, gaming, augmented reality, virtual reality, and artwork creation, assets are often created in styles such that they altogether harmonize to create an intended look and feel of the final scenes. A common practice is to first create the assets, and then post-process them to match some styles, often known as *stylization*. The advent of modern deep learning has led to the emergence of *neural stylization*, a family of methods that automatically create visual content in styles, facilitating the exploration of aesthetics in the creation of visual data. Neural stylization is applicable to visual data in general, including images, videos, and 3D data. Unlike image stylization, which has been well developed in the past decade, neural stylization for 3D data remains an open area to explore for new creative vistas and practical applications.

Communicated by Shengfeng He.

✉ Yingshu Chen
ychengw@connect.ust.hk

Guocheng Shao
gshao@connect.ust.hk

Ka Chun Shum
kcshum@connect.ust.hk

Binh-Son Hua
binhson.hua@tcd.ie

Sai-Kit Yeung
saikit@ust.hk

[1] Hong Kong University of Science and Technology, Sai Kung District, Hong Kong

[2] Trinity College Dublin, Dublin, Ireland

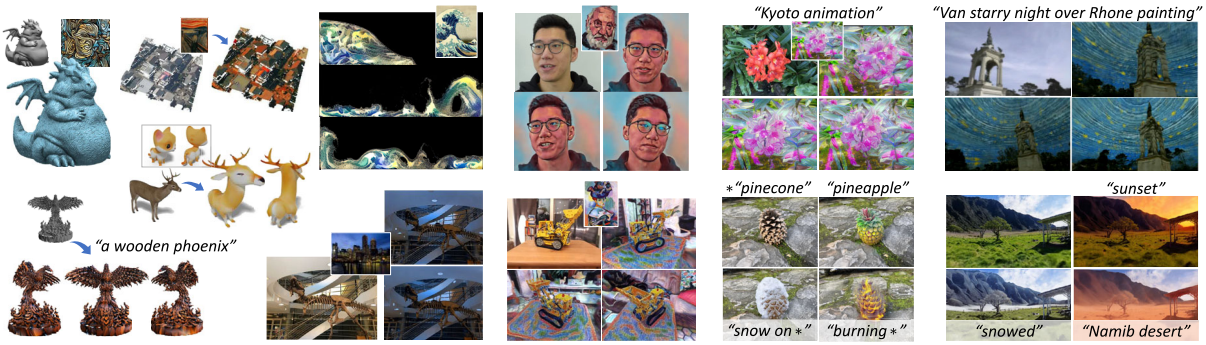


Fig. 1 The survey delves into the realm of neural stylization on diverse 3D representations, including meshes, point clouds, volume, and neural fields. The neural stylization with visual, textual and geometric features retrieved from large-scale neural models empowers artistic, photorealistic, and semantic style transformation of the geometry and appearance of 3D scenes. Images adapted from Liu et al. (2018), Ma et al. (2023), X. Cao et al. (2020), Yin et al. (2021), Wang et al. (2023b), Zhang et al. (2022), Zhang et al. (2023c), Song et al. (2023), Haque et al. (2023)

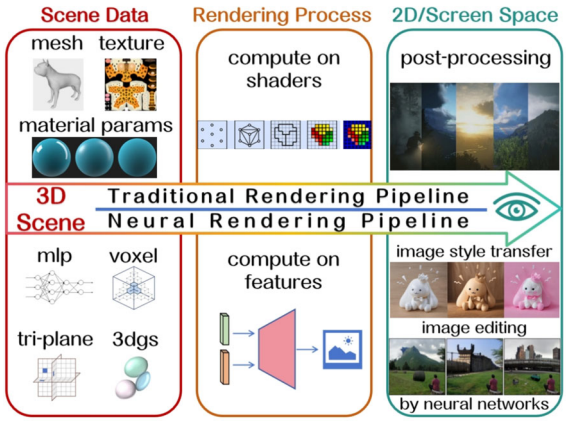


Fig. 2 A general overview of mesh-based and radiance field-based rendering pipelines. Images adapted from Yin et al. (2021), G. Kim et al. (2024), Chen and Wang (2024b), Avrahami et al. (2022), Spline (2023)

This report delves into the latest developments in the creation of 3D digital art using neural stylization methods, as shown in Fig. 1. Neural stylization can facilitate automated design and creation of explicit meshes, textures and volumetric assets, which supports seamless usage in the traditional rendering pipeline, accelerating labor-intensive manual tasks such as modeling, texturing, and simulation. Neural stylization also enables efficient and controllable manipulation or transformation of neural scenes, which typically utilizes neural networks instead of shaders to generate images (Fig. 2). Interestingly, neural stylization has shown practical importance in various applications, including 3D texture design and artistic simulation in movie making (Navarro & Rice, 2021; Kanyuk et al., 2023; Hoffman et al., 2023), virtual production (Manzaneque, 2023), mixed reality experiences (Tseng et al., 2022; Taniguchi, 2019), and artwork creation (Guljajeva & Canet, 2022).

Despite its advantages, performing neural stylization in 3D presents new technical challenges such as multi-view consistency, view sampling and rendering, as well as robustness issues including the relative scarcity of 3D datasets, and memory consumption for training and inference. This report provides a comprehensive discussion and summary of advanced 3D neural stylization techniques that address these challenges, highlighting the power of neural rendering (Tewari et al., 2022), vision-language models (Radford et al., 2021), and large-scale generative models (Rombach et al., 2022).

The structure of this report is depicted in Fig. 3, which is outlined as follows: Sect. 2 reviews fundamentals of 2D neural stylization and important visual or textural feature priors, which act as components or backbones of 3D neural stylization techniques. In Sect. 3, we introduce a taxonomy for neural stylization and discuss advanced stylization methods on various types of 3D representations in depth with summaries of practical tips to guide future works. In Sect. 4, we summarize popular datasets and evaluation metrics for 3D stylization, and particularly, we deliver a benchmark of 3D neural stylization to serve as a reference for the performance of selected methods. Sect. 5 introduces diverse applications of 3D neural stylization, demonstrating its practical value in a wide range of domains. Finally, Sect. 6 highlights promising research directions with practical significance.

1.1 Definition and Terminologies

Definition 1 *3D neural stylization* refers to the process that employs deep learning techniques and stylization algorithms to generate stylized 3D digital assets or the stylized rendering from these assets, including the alteration of appearance and/or geometry.

3D neural stylization is well connected to the following terminologies and techniques.

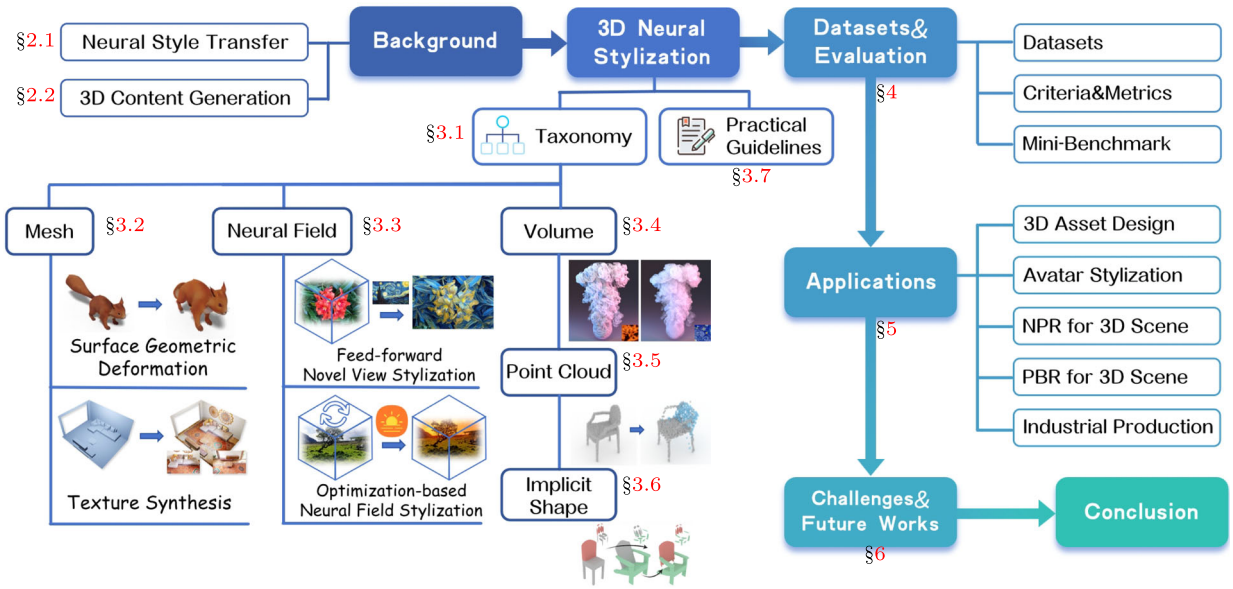


Fig. 3 Structure of our survey

• *Neural Style Transfer (NST)* refers to a class of algorithms that manipulate digital images, or videos, in order to adopt the appearance or visual style of target reference while preserving original content features. It can be regarded as a foundation of 3D neural stylization, as most image-guided 3D stylization methods rely on it. We refer to former surveys (Jing et al., 2019; Singh et al., 2021; Zhan et al., 2023) and provide a concise review in Sect. 2.1.

• *Neural Rendering* is a class of techniques that "learn to render and/or represent a scene from real-world imagery, which can be an unordered set of images, or structured, multi-view images or videos" (Tewari et al., 2022), several works of which focus on the generation of photorealistic rendering from the neural representations. Instead, neural stylization aims at modifying the visual appearance and aesthetic characteristics of existing digital representations and obtaining artistic or photorealistic rendering results. We refer readers to existing surveys for insight into neural rendering (Tewari et al., 2020, 2022), neural fields (Xie et al., 2022) and image generation (Zhan et al., 2023).

• *Non-photorealistic Rendering (NPR)* is an area of computer graphics that enables abstract stylized rendering for either 3D models, 3D images or 2D images, such as toon shading, Gooch shading (Gooch et al., 1998), stroke-based painterly rendering (Haeberli, 1990; Hertzmann, 1998), patch-based texture synthesis and transfer (Efros & Freeman, 2001; Hertzmann et al., 2001). Mainly leveraging programmable shaders and image-processing techniques, NPR has been widely used in the realms of animation making, digital content creation (Artineering, 2018) and game development (McGuire et al., 2010). However, these techniques require creating handcrafted style patterns and rules, which are labor-intensive and entail domain expertise. In contrast, neural stylization enables fast production with arbitrary style references and has been applied to accelerate cinematic digital production (Joshi et al., 2017; Navarro & Rice, 2021; Hoffman et al., 2023).

• *Neural Scene Editing* has become more practical in the recent few years thanks to the contribution of large language models (LLMs) and vision-language models (VLMs) (Radford et al., 2021; Li et al., 2023a). Editing methods focus on adding, modifying, or removing objects in a scene, or manipulating some regions of interest. By contrast, stylization methods focus on the transfer of overall appearance and the adoption of specific aesthetic characteristics. Still, stylization methods share critical ideas with editing methods, and some of the methods covered in this survey can also apply to scene editing (Koo et al., 2023; Song et al., 2023; Bao et al., 2023).

1.2 Related Surveys

In the literature, there exist comprehensive surveys on 2D neural style transfer (Jing et al., 2019; Singh et al., 2021), surveys on generative image models (Zhan et al., 2023; Croitoru et al., 2023; Yang et al., 2023), and surveys on neural field representations (Xie et al., 2022) and rendering (Tewari et al., 2020, 2022). Our survey aims to explore the potential of connecting neural stylization techniques with both traditional and advanced 3D representations, thereby offering valuable resources for style-based 3D digital designs. To the best of our knowledge, this paper is the first comprehensive review to summarize neural stylization techniques and applications

specifically tailored to 3D data, highlighting the immense capabilities of neural stylization in the 3D domain.

2 Background

In this section, we provide a brief discussion on neural style transfer for images, which serves as the fundamental building block for discussing 3D neural stylization (Sect. 2.1). This covers techniques leveraging visual or textual guidance for image style transfer and manipulation, as well as insights on linkages to 3D stylization domain. We also discuss generic methods for 3D content generation with a focus on the state-of-the-art diffusion models for 3D generation (Sect. 2.2).

2.1 Neural Style Transfer

The basic idea of neural style transfer is to reproduce the style of a reference image for an input image, while keeping the original content of the input. The content representation of an image can be extracted by predicting its features using a pre-trained or trainable encoder (Simonyan & Zisserman, 2015; Huang et al., 2018). The style representation can be represented by a Gram matrix, which is a dot-product matrix measuring the relevance of each pair of features extracted by a pre-trained network (Gatys et al., 2016). Alternatively, the style of an image can also be characterized by the spatially invariant statistics (i.e. channel-wise mean and variance) of features (Dumoulin et al., 2017; Huang & Belongie, 2017). With the rise of vision-language pre-trained models, textual embeddings have been widely employed to represent content or style information (Radford et al., 2021; Li et al., 2023a). In Fig. 4, we provide a high-level pipeline comparison of different types of neural style transfer methods, including single-style transfer via optimization, arbitrary style transfer via feed-forward network, and style transfer via generative models. We discuss each type of method below.

2.1.1 Single Style Transfer

One simple method for neural style transfer (Fig. 4a) is to optimize from a white noise image to obtain a new image that shares the content of a source image and the style of a reference image (Gatys et al., 2016). Given a content source image c and a style reference image s, the optimization can be done by minimizing a combined objective: $\mathcal{L}_{total} = \mathcal{L}_c(c, cs) + \lambda \mathcal{L}_s(s, cs)$, where the total loss consists of a content loss $\mathcal{L}_c$ of the squared-error of VGG features between content image c and output stylized image cs, and a Gram matrix style loss $\mathcal{L}_s$ between style image s and output stylized image. λ is a hyperparameter.

Instead of optimizing an image for each transfer, we can train a single feed-forward network with a perceptual loss to perform style transfer for arbitrary content images (Johnson et al., 2016). At inference, real-time stylization can be performed simply by forwarding an arbitrary content image through the network. Although performing network inference is much faster than running an optimization, one still needs to retrain the network for each different style.

The rise of vision-language models (Radford et al., 2021) leads to the possibility of performing style transfer using text prompts as guidance. Given CLIP with a text encoder and an image encoder sharing the same latent embedding space (Radford et al., 2021), text-guided image style transfer can be achieved by maximizing text-image semantic similarity as the style loss, usually formulated by the CLIP loss defined by

$$\mathcal{L}_{clip}(cs, s_{sty}) = 1 - sim(E_I(cs), E_T(s_{sty})), \tag{1}$$

where cs and s_{sty} are the stylized image and style text prompt, E_I and E_T are the pre-trained CLIP image encoder and text encoder, respectively. $sim(A, B)$ is the cosine similarity between two feature vectors. One can also use the directional CLIP loss (Patashnik et al., 2021; Gal et al., 2022) to achieve better style transfer quality:

$$\mathcal{L}_{dir}(c, cs, s_{src}, s_{sty}) = 1 - sim(\Delta I, \Delta T), \tag{2}$$

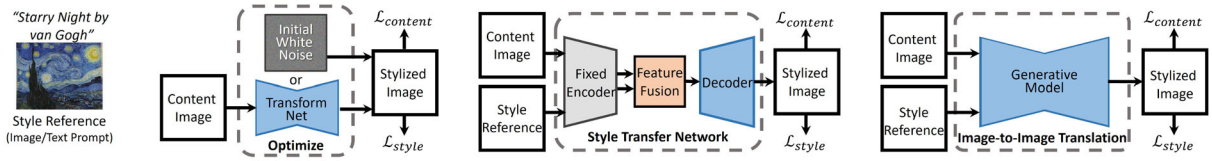
where $\Delta I = E_I(cs) - E_I(c)$, $\Delta T = E_T(s_{sty}) - E_T(s_{src})$, c is the content image, cs is the stylized image, s_{src} and s_{sty} are the source (content) style prompt and target style prompt. An example of s_{src} and s_{sty} can be "Photo" and "Picasso style painting", respectively (Kwon & Ye, 2022). Since CLIP does not support high-resolution image embedding, a patch-wise version of the directional CLIP loss with augmented patches can be used for better artistic semantic texture transfer (Kwon & Ye, 2022).

2.1.2 Arbitrary Style Transfer: AdaIN and LST

To enable the model to transfer arbitrary styles without retraining, one can employ an autoencoder with style fusion (Fig. 4b). Particularly, to fuse the content and style features, we can use adaptive instance normalization (AdaIN) that directly regulates the mean and variance of the feature maps of the content image to match those of the target style image (Huang & Belongie, 2017):

$$AdaIN(c, s) = \sigma(F(s)) \left(\frac{F(c) - \mu(F(c))}{\sigma(F(c))} \right) + \mu(F(s)), \tag{3}$$

where each VGG feature map $F(\cdot)$ is normalized separately. The transformed feature maps are fed into a learned decoder to generate the final output.



(a) Single-style optimization (b) Arbitrary style transfer (c) Image-to-image translation

Fig. 4 Pipeline comparisons of 2D neural style transfer. **a** Single-style transfer via optimization (Gatys et al., 2016; Kwon & Ye, 2022; Johnson et al., 2016). **b** Arbitrary style transfer via feature fusion or transformation (Huang & Belongie, 2017; Li et al., 2019; Liu et al., 2021). **c** Image-to-image translation with style condition via generative models (Huang et al., 2018; Deng et al., 2022; Wen et al., 2023; Zhang et al., 2023d)

Alternatively, we can fuse content and style features by learning an affine feature transformation matrix T from content features $F(c)$ and style features $F(s)$ through a convolutional neural network, as proposed by linear style transfer (LST) (Li et al., 2019):

$$LST(c, s) = T \cdot (F(c) - \mu(F(c))) + \mu(F(s)). \tag{4}$$

2.1.3 Generative Models for Style Transfer

In addition to optimization-based and feed-forward inference methods, neural style transfer can be achieved via image synthesis tasks such as image-to-image translation and generative models. Particularly, image-to-image translation (I2I) achieves style transfer by translating an image from one source domain to a target domain using a generative model such as a generative adversarial network (GAN) (Goodfellow et al., 2014) (Fig. 4c). Deterministic I2I translation methods focus on task-specific domain-to-domain translation and do not require style guidance via reference images (Isola et al., 2017; Zhu et al., 2017; Liu et al., 2017). Multi-modal I2I translation models enable translation based on examples or latent features (Huang et al., 2018; Lee et al., 2018; Chang et al., 2020; Chen et al., 2022a). While GAN-based I2I models generate high-fidelity images, they are domain-specific and resource-consuming for training compared to traditional methods (Fig. 4a, b).

Recently, diffusion models have demonstrated state-of-the-art performance for image synthesis (Rombach et al., 2022). A particular strength of diffusion models that contributes to their wide adoption is their ability to learn across different data modalities. Similarly to the spirit of CLIP loss for style transfer, text-guided diffusion models leverage textual embedding of the text prompt for conditional image generation and thus allow style transfer via text-to-image generation and text-guided I2I translation (Rombach et al., 2022; Saharia et al., 2022; OpenAI, 2023). Among the publicly available text-to-image diffusion models, the Stable Diffusion series (Rombach et al., 2022; Podell et al., 2023; Esser et al., 2024) is the most representative. They have inspired a vast amount of research work and a broad range of downstream applications.

Numerous works explored the potential of text-to-image diffusion models for style transfer. One track fine-tunes the diffusion model (usually U-Net) or learns a special textual embedding using a set of images with target style, including techniques like Dreambooth (Ruiz et al., 2023), LoRA (Hu et al., 2022; Frenkel et al., 2024), textual inversion (Gal et al., 2022; Zhang et al., 2023d), etc. Among them, B-LoRA (Frenkel et al., 2024) jointly fine-tunes two blocks of LoRA layers to capture respectively the style and content of an image, enabling the transfer of this style to unseen content, or such content to new styles. Textual inversion method InST (Zhang et al., 2023d) binds a special language mark with a textual embedding inversed from a style image. This style can then be transferred to other images by including the language mark in the prompt during inference.

Another track leverages attention modules in diffusion U-Nets to embed style information without per-style optimization. Spatially invariant feature statistics, as discussed in Sect. 2.1.1, represent style effectively. Diffusion in Style (Everaert et al., 2023) pre-computes the mean and variance for Gaussian noise sampling based on the style feature statistics. StyleID (Chung et al., 2024f) employs AdaIN for noise initialization and replaces content's self-attention *key, value* with those from the style layers. StyleAlign (Hertz et al., 2024) further applies AdaIN to the *query, key* of a sequence of generated images to ensure style consistency. DEADiff (Qi et al., 2024) focuses on cross-attention in high-resolution layers, utilizing Q-Former (Li et al., 2023a) for style extraction instead of AdaIN. InstantStyle (Wang et al., 2024) isolates styles for transferring by subtracting content CLIP embeddings from their image CLIP embeddings.

Moreover, large pre-trained image editing models such as Instruct-Pix2Pix (Brooks et al., 2023) showcase good style transfer capability. Leveraging LLMs, e.g., GPT-3 (Brown et al., 2020), Instruct-Pix2Pix automates the generation of diverse text prompts and adopts Prompt-to-Prompt (2023b) to create corresponding image pairs. Subsequently, a standard diffusion model is trained on these pairs, with the

exception of concatenating the source image to the first network layer as conditional information. However, its performance degrades when intricate styles are difficult to express in natural language.

2.1.4 Linking 2D to 3D Stylization

The exploration of 2D neural style transfer offered us valuable insights into style feature and conversion: *spatially invariant statistics (mean and variance) of visual feature maps can represent the image style; one can shift such statistics (to align with those of other images) to control the style of an image*. This insight and several other practical skills can boost 3D stylization research. We briefly exemplify three directions of 3D stylization below.

1. *Loss Function Design.* Sect. 2.1.1 revisited the basic version of loss functions for 2D style transfer tasks. When it comes to 3D stylization, it is straightforward to employ similar optimization losses in a view-by-view manner while maintaining multi-view consistency via 3D representation. For instance, Liu et al. (2018) apply 2D latent content and style losses (Gatys et al., 2016) to supervise mesh surface morphing; Chen et al. (2024e) optimize texture style jointly with semantics-aware target image features (mean and standard deviation) and textual features (Eq. 1-2).
2. *Feed-forward Feature Transform.* The success of feed-forward arbitrary style transfer through feature statistics transformation in the 2D domain inspires feed-forward 3D neural style transfer with 3D-aware feature representation. For example, StyleGaussian (Liu et al., 2024) applies AdaIN (Eq. 3) for VGG (Simonyan & Zisserman, 2015) features stored in 3D Gaussians for efficient 3D style transfer. FPRF (Kim et al., 2024) applies a semantics-aware local AdaIN for features stored with tri-planes. StyleRF (Liu et al., 2023) proposes a modified volume-adaptive IN for features obtained from feature grids.
3. *Stylization with Generative Priors.* The burgeoning 2D large generative models (Sect. 2.1.3) have been leveraged to handle the stylization and even address 3D consistency issues (with geometry priors). A simple yet effective way is to directly stylize a sampled view as the target using pre-trained generative models, followed by backpropagation of the 2D error, such as IN2N (Haque et al., 2023) and IG2G(Vachha & Haque, 2024). A more advanced approach is score distillation, which leverages the capability of diffusion model to process multi-modal guidance for better controllability. Score distillation was first proposed and widely adopted in 3D generation tasks. We'll discuss the topic in the next part.

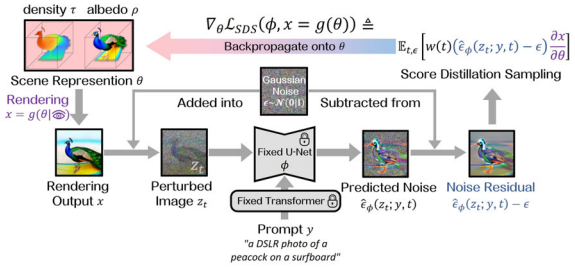


Fig. 5 3D generation architecture with score distillation sampling loss. A pre-trained denoising U-Net supervises NeRF optimization. Image adapted from Poole et al. (2023)

2.2 3D Content Generation

Equipped with a background in neural style transfer, let us now briefly discuss 3D content generation methods, which serve as the background and provide valuable insights for 3D neural stylization subsequently.

3D Representations In contrast to image representations, there are various representations for learning to generate 3D content. Conventional 3D representations are mostly explicit representations including triangle and polygon meshes, point clouds, and voxel grids (or volumes). The advances in deep learning have spurred an increasing interest in using neural networks to represent 3D data as neural fields, notably neural radiance fields (NeRF) (Mildenhall et al., 2020). Subsequently, there appeared notable hybrid or compact radiance fields representations, featured by neural graphics primitives (NGP) (Müller et al., 2022) and 3D Gaussian splats (3DGS) (Kerbl et al., 2023). Some implicit representations, such as signed distance functions (SDF) and their truncated versions (TSDF), also gain popularity to represent implicit shapes. We refer readers to the existing survey for a comprehensive overview of neural fields for visual computing (Xie et al., 2022).

3D Generative Models Existing 3D generative models have explored different types of 3D representations such as point clouds, voxel grids, meshes, and implicit fields (Zhao et al., 2021; Qi et al., 2017; Wu et al., 2015; Masci et al., 2015; Chen & Zhang, 2019). 3D data-driven generative models are trained with large-scale 3D assets with diverse appearances and shapes, which are challenging to collect (Chang et al., 2015; Deitke et al., 2023; Liu et al., 2019). Inspired by neural volume rendering, there appeared a group of 3D-aware image synthesis works learning 3D generation from accessible 2D data (Niemeyer & Geiger, 2021; Nguyen-Phuoc et al., 2020; Chan et al., 2022; Gu et al., 2022). Since the slow and resource-intensive nature of volume rendering results in long training time and low resolution, one can leverage a reduced 3D representation such as tri-planes in the GAN framework for efficient and high-quality image and 3D data generation (Chan et al., 2022).

3D Generation via Diffusion Priors

In a similar spirit to 3D-aware image synthesis, it is of great interest to generate 3D data from priors learned by 2D diffusion models (OpenAI, 2023; Ruiz et al., 2023). DreamFusion (Poole et al., 2023) proposed a 3D generation pipeline that optimizes a NeRF by leveraging a text-guided diffusion model as the critic to the NeRF rendered images (Fig. 5). The gradient function for optimization is a weighted average of noise residual multiplied by the Jacobian of the rendering process, also known as score distillation sampling (SDS).

Several variants have been proposed to resolve serious problems in the SDS loss such as oversaturation, oversmoothing, and lack of details, generating more realistic and high-definition 3D objects. Variational score distillation (VSD) trains a LoRA model to better estimate the data distribution of rendered images for effective updating (Wang et al., 2023c). Delta denoising score (DDS) computes the difference between two SDS scores as guidance (Hertz et al., 2023a), while posterior distillation sampling (PDS) aligns the stochastic latents of the source image and the target image instead of noise variables (Koo et al., 2024). Further works explore instilling geometric information to score distillation (Yang et al., 2023; Yeh et al., 2024).

Moreover, 3D datasets are also exploited to provide geometric priors such as canonical coordinates map (Li et al., 2024a) and normal map (Long et al., 2024) for better multi-view consistency. Zero-1-to-3 (Liu et al., 2023) proposed the view-conditioned diffusion that accepts a rotation angle as an extra condition, which synthesizes novel views based on any single view of a 3D model. Recent 3D generation works further improve the visual quality by combining 2D priors from text-to-image diffusion and 3D-aware priors from view-conditioned diffusion (Qian et al., 2024; Sun et al., 2024).

3 3D Neural Stylization

In this section, we first establish a taxonomy for neural stylization and give an example of the categorization of selected 3D neural stylization methods (Sect. 3.1). In the subsequent sections, we will discuss state-of-the-art 3D neural stylization techniques on diverse 3D representations, such as meshes (Sect. 3.2), neural fields (Sect. 3.3), volumetric data (Sect. 3.4), point clouds (Sect. 3.5), and implicit shapes (Sect. 3.6). We then discuss a set of guidelines for practical implementations of 3D stylization (Sect. 3.7).

3.1 Taxonomy

Our taxonomy for neural stylization methods consists of the following aspects:

- *Representations*. We categorize stylization methods based on data representations such as image, mesh, volume, point cloud, and neural field.
- *Neural Style Feature*. We categorize based on image visual features, textual semantic features, or 3D latent features derived from pre-trained models, typically neural classifiers or generative models.
- *Optimization*. This refers to optimization-based or prediction-based stylization methods with single, multiple, or arbitrary styles supported.
- *Stylization Genres*. This refers to different types of stylization, mainly including geometry stylization operating on asset shape and surface patterns, and appearance stylization focusing on color, texture and visual patterns to align with specific styles from artistic paintings to realistic concepts.

To guide the reader through the main section of this survey, we illustrate the taxonomy in Fig. 6, and provide a hierarchical classification of the 3D stylization methods in Fig. 7. Let us now discuss 3D neural stylization methods by following the categorization based on 3D representations below.

3.2 Mesh-Based Stylization

In computer graphics and 3D modeling, a mesh is a collection of vertices, edges, and faces that define the geometric structure of an object. Objects represented by meshes can also store additional appearance attributes, such as vertex colors, materials, UV coordinates, and texture maps. Additionally, neural networks, such as multi-layer perceptrons (MLPs), can represent these attributes, including neural textures (Thies et al., 2019; Oechsle et al., 2019), neural reflectance field (Baatz et al., 2022), neural visibility field (Srinivasan et al., 2021), neural vertex (Michel et al., 2022; Ma et al., 2023; Lei et al., 2022), etc. By using differentiable renderers (Ravi et al., 2020; Laine et al., 2020; Fuji Tsang et al., 2022), we can optimize these explicit or implicit attribute representations for 3D geometry manipulation and appearance editing. For example, one can predict vertex positions and colors (Michel et al., 2022), SVBRDF parameters and normals (Lei et al., 2022), and synthesize new texture images (Richardson et al., 2023). The following sections cover critical techniques for mesh-based stylization, including geometric deformation (Sect. 3.2.1) and texture synthesis (Sect. 3.2.2) to align with a provided image, text, or 3D shape guidance. Table 1 shows a comparison of recent mesh-based stylization methods.

3.2.1 Surface Geometric Deformation

3D neural stylization enables deforming mesh geometry to align with artistic visual patterns or a specified shape, guided

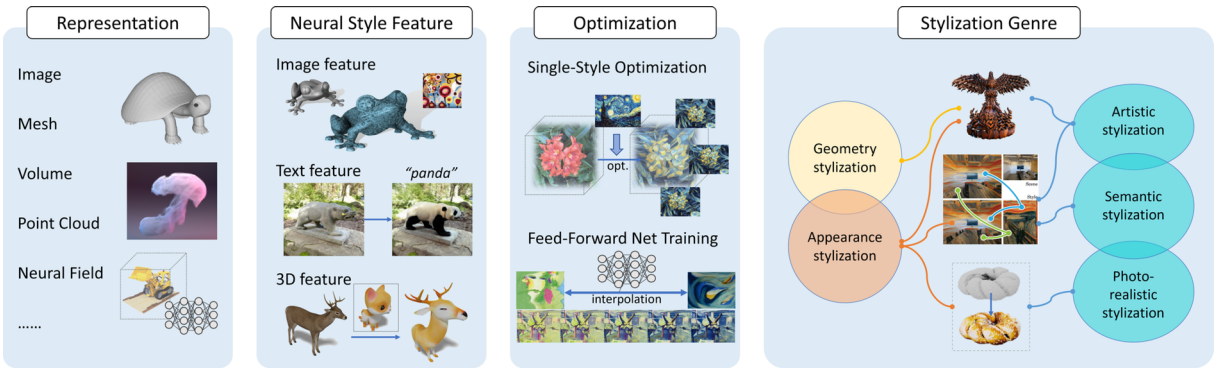


Fig. 6 Taxonomy of neural stylization. Images from Richardson et al. (2023), Aurand et al. (2022), Mildenhall et al. (2020), Liu et al. (2018), Haque et al. (2023), Yin et al. (2021), Zhang et al. (2022), Liu et al. (2023), Chen et al. (2023a), Pang et al. (2023)

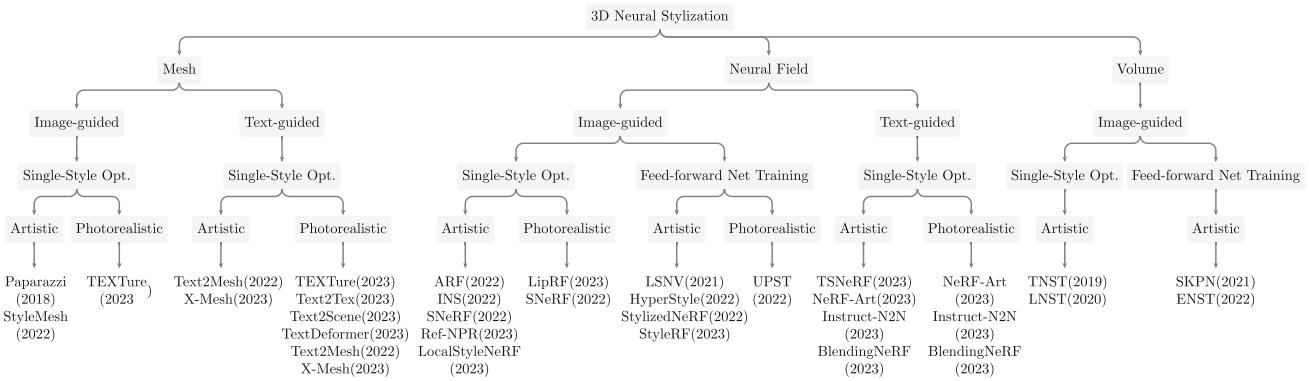


Fig. 7 Hierarchical classification of selected image- and text-guided 3D neural stylization methods

by visual references or textual descriptions. This capability facilitates creative 3D modeling such as surface engraving effect (Liu et al., 2018) and geometry morphing (Gao et al., 2023). Existing works learn geometry variations in the form of vertex position displacement, e.g., explicit displacement via differentiable rendering (Liu et al., 2018) and implicit displacement via neural networks (Michel et al., 2022; Ma et al., 2023; Gao et al., 2023).

For example, Paparazzi is a neural stylization method based on differentiable rendering that allows the propagation of changes in the image domain to changes of the mesh vertex positions (Liu et al., 2018). It takes a triangle mesh as input, and applies latent VGG content and style losses (Sect. 2.1, Gatys et al. (2016)) between rendered image(s) and gray-scale style image to update vertex positions. After convergence, the mesh surface is stylized with artistic strokes and motifs from the style image.

With CLIP loss (Sect. 2.1.1, Eq. 1), recent works explored text-guided mesh geometric and/or appearance alteration (Michel et al., 2022; Ma et al., 2023). Text2Mesh (2022) and X-Mesh (2023) incorporate Neural Style Field, which is composed of an MLP network that maps vertex coordinates to vertex color (offset) and vertex position offset. The stylized mesh, with updated vertex colors and vertex positions, is rendered into multiple colored and gray-scale images, which are used for computing CLIP loss against a given text prompt.

Besides updating with the CLIP loss, X-Mesh (Ma et al., 2023) employs an attention module to directly include the prompt CLIP embedding as additional input along with the vertex coordinate embeddings. Accordingly, X-Mesh achieves fast convergence in a few minutes for high-quality stylized results. TextDeformer (Gao et al., 2023) upgraded the local CLIP-guided mesh geometric stylization (Wang et al., 2022; Michel et al., 2022) through Jacobians for global and smooth mesh deformation (Aigerman et al., 2022). Instead of learning position displacement directly, they assign Jacobians by matrices for each triangle and solve a Poisson problem (Aigerman et al., 2022) to compute the corresponding vertex deformation map, which largely achieves deformation with low-frequency to high-frequency details.

Alternatively, 3DStyleNet (Yin et al., 2021) learns to perform joint geometric and texture style transformation from one 3D object to another, and interpolation of geometric and texture style. This method consists of a 3D geometric part-aware style transfer network and a 2D texture style transfer network. The authors innovatively abstracted the geometries of an object with a set of 3D Gaussian ellip-

Table 1 Summary of selected mesh-based neural stylization. PBR refers to physically-based rendering

Method	Input	Guidance	Output	Scene	Optimization	Style Genre	⟨/⟩
3DStyleNet (Yin et al., 2021)	Textured mesh	Points, image	Textured mesh	Object	Feed-forward	Geo&app.	✗
Point-UV Diffusion (Yu et al., 2023)	Mesh	Points, image or text	Textured mesh	Object	Feed-forward	Appearance	✓
TexDreamer (Liu et al., 2024)	Human mesh	Image or text	Texture	Human	Feed-forward	Appearance	✓
Paparazzi (Liu et al., 2018)	Mesh	Image	Mesh	Object	Single-style	Geometry, artistic	✓
StyleMesh (Höllein et al., 2022)	RGBD imgs, rec. mesh	Image	Texture	Room	Single-style	appearance, artistic	✓
Text2Mesh (Michel et al., 2022)	Mesh	Text or image	Vertex-colored mesh	Object	Single-style	geo&app.	✓
X-Mesh (Ma et al., 2023)	Mesh	Text	Vertex-colored mesh	Object	Single-style	geo&app.	✓
Text2Scene (Hwang et al., 2023)	Meshes, segment labels	Image & text	Vertex-colored mesh	Room w/objects	Single-style	Appearance	✓
TextDeformer (Gao et al., 2023)	Mesh	Text	Mesh	Object	Single-style	Geometry	✓
TEXTure (Richardson et al., 2023)	Mesh	Text or image	Texture	Object	Single-style	Appearance	✓
Text2Tex (Chen et al., 2023a)	Mesh	Text	Texture	Object	Single-style	Appearance	✓
TexFusion (Cao et al., 2023)	Mesh	Text	Texture	Object	Single-style	Appearance	✗
3DStyle-Diffusion (Yang et al., 2023)	Mesh	Text	Neural texture	Object	Single-style	Appearance	✓
SceneTex (Chen et al., 2024a)	Mesh	Text	Texture	Room	Single-style	Appearance	✓
Paint-it (Youwang et al., 2024)	Mesh	Text	PBR texture	Object	Single-style	Appearance	✓
Paint3D (Zeng et al., 2024)	Mesh	Image or text	Texture	Object	Single-style	Appearance	✓
TeMO (Zhang et al., 2024e)	Multi-meshes	Text	Textures	Objects	Single/multi-style	Appearance	✓
EASI-Tex (Perla et al., 2024)	Mesh	Image	Texture	Object	Single-style	Appearance	✓
DreamMat (Zhang et al., 2024f)	Mesh	Text	PBR texture	Object	Single-style	Appearance	✓
FlashTex (Deng et al., 2024)	Mesh	Text	PBR texture	Object	Single-style	Appearance	✓
StyleCity (Chen et al., 2024e)	Textured mesh	Image & Text	Texture	City	Single-style	Appearance	✓
ControllableNST (Haetinger et al., 2024)	Static/dynamic mesh	Image	Static/dynamic (colored) Mesh	Object(s)	Single-style	Geo.(&app.)	✗

soids and employed a learned 3D part-aware semantics affine transformation field based on Linear Blend Skinning (LBS) model (Lindholm et al., 2001). Meanwhile, they transfer the mesh texture using regular image style transfer techniques (Sect. 2.1 LST (2019)). The two networks are pre-trained with non-textured mesh models (TurboSquid, 2023; Chang et al., 2015; Renderpeople, 2023) and images (WikiArt (2016) and COCO (2014)) respectively. They are then jointly optimized by part-aware, content and style losses through rendered multi-views via a differentiable renderer (Laine et al., 2020).

A recent work (Haetinger et al., 2024) further explores geometry stylization for dynamic meshes, enabling efficient production of physics simulation and animation. They employ neural neighbor style transfer (Kolkin et al., 2022) instead of Gram-matrix to guide the style transfer, which produces higher quality high-frequency details by replacing each individual feature of the content image with its closest feature of the style image. The key to an effective and natural global and local stylization is the multi-level parameterization of mesh vertex position, which allows the 2D error to propagate in differential rendering sufficiently. An additional mechanism for vertex displacement interpolation and smoothing across frames is applied to improve time coherency. These enhancements result in high-quality, artifact-free mesh stylizations, suitable for creating unique artistic looks in simulations and 3D asset design.

3.2.2 Texture Synthesis

Mesh textures are essential for representing complex visual appearance with color and patterns, for which image- or text-guided neural style transfer are well developed (Sect. 2.1). Several methods have explored texture transformation and synthesis using visual features such as VGG features and diffusion priors (Höllein et al., 2022; Lei et al., 2022; Richardson et al., 2023; Cao et al., 2023; Yang et al., 2023). We categorize these methods based on their optimization and learning techniques and discuss them below.

Optimize via 2D Features With an artistic image reference, StyleMesh (Höllein et al., 2022) proposed a depth- and angle-aware texture optimization scheme for the 3D reconstructed indoor room. It optimizes an explicit texture image by back-propagating gradients computed from 2D content and style losses (Sect. 2.1, Gatys et al. (2016)) between each view of the scene and the style reference image. This method leverages depth and normal information from the mesh, mitigating artifacts such as view-dependent stretch and size artifacts that commonly arise from conventional 2D losses in 3D scenarios, as shown in Fig. 8. Nonetheless, StyleMesh largely relies on posed images under reconstructed scenes and ground-truth depths.

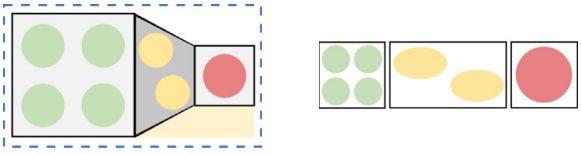


Fig. 8 Stretched pattern artifacts from stylization in screen space. Image adapted from Kato et al. (2018)

Optimizing mesh appearance colors with style descriptions using CLIP is effective but may not always achieve realistic results (Michel et al., 2022; Lei et al., 2022; Ma et al., 2023). Recently, text-to-image diffusion models have gained popularity for their ability to synthesize high-fidelity images. Therefore, researchers start to explore lifting 2D diffusion priors for 3D generation (Poole et al., 2023; Wang et al., 2023c; Lin et al., 2023; Chen et al., 2023b) and stylization (Chen et al., 2023a; Yang et al., 2023; Zeng et al., 2024; Youwang et al., 2024). Among these, TEXTure (Richardson et al., 2023) is a text-guided 3D texture painting method that iteratively paints the texture in a view-by-view manner. To maintain 3D consistency, each view drawing iteration is guided by a view-dependent trimap that indicates the “keep", “refine", and “generate" regions to control the amount of newly generated content for the texture. Alongside the trimap, a rendered RGB and depth map are fed into a pre-trained depth-to-image diffusion model (Sect. 2.1.3, ControlNet (2023b)) to obtain a synthesized view. This synthesis is finally projected back to the texture map via optimization. Apart from texturing, TEXTure supports various tasks such as texture transfer, texture editing, and multi-view image transfer.

A concurrent work to TEXTure is Text2Tex (Chen et al., 2023a), which inpaints likewise the texture images from different views progressively with the help of the confident trimap and a depth-to-image ControlNet (2023b). This method presets some axis-aligned viewpoints and alternatively updates the next best view (see Table 4), which follows a more robust automatic view scheduling strategy addressing the blurriness and stretching artifacts. Paint3D (Zeng et al., 2024) employs coarse-to-fine UV diffusion models to further refine incomplete areas of multi-view inpainted texture in high definition. Based on an indoor room scenario, DreamSpace (Yang et al., 2024) synthesizes an indoor panorama with additional inpainting with diffusion models (Zhang et al., 2023b) for more consistent texture synthesis.

Optimize in Latent Space While running the entire generative diffusion process for multi-view painting provides an efficient approach for texture synthesis, it often results in inconsistent texture patterns and overall style. Instead, TexFusion (Cao et al., 2023) updates a 3D consistent latent

texture at each denoising step from multi-views conditioned on previous denoising steps. To ensure consistency, the final texture image is optimized by distilling multi-view images decoded by a pre-trained depth-conditioned diffusion model (Sect. 2.1.3, Stable Diffusion (2022)) and with a neural color field mapping 3D coordinates to RGB values (Müller et al., 2022). Similarly, Knodt and Gao (2023) updates a latent texture map from multi-view via MultiDiffusion (Bar-Tal et al., 2023), a multi-window joint diffusion technique for multi-view consistency.

Optimize via Score Distillation Inspired by score distillation sampling (SDS) techniques (Sect. 2.2, Fig. 5), several works employ SDS and its variants for texture optimization, focusing on a single object or a single room (Yang et al., 2023; Guo et al., 2023; Wu et al., 2023; Chen et al., 2024a; Yeh et al., 2024). In particular, TextureDreamer (Yeh et al., 2024) and 3DStyle-Diffusion (Yang et al., 2023) adopt a neural field representation with BRDF parameters (Lei et al., 2022) to facilitate photorealistic rendering. Both works (Yang et al., 2023; Yeh et al., 2024) incorporate geometry-conditioned score distillation from ControlNet (2023b), leveraging additional inputs such as depth, normal, and camera pose. Additionally, Decorate3D (Guo et al., 2023) and HyperDreamer (Wu et al., 2023) utilize super-resolution diffusion techniques to enhance the synthesis of textures at higher resolutions.

Optimize with 3D Shape Supervision Point-UV Diffusion (Yu et al., 2023) explores texture synthesis leveraging the shape attributes such as vertex coordinates, normals, and segment masks of the mesh model. The proposed coarse-to-fine texture synthesis framework that combines a point diffusion network (Liu et al., 2019; Zhou et al., 2021) and a UV diffusion network enables unconditioned texture synthesis for arbitrary mesh models of each training category in the ShapeNet dataset (Chang et al., 2015). The pipeline can also receive additional image or text guidance using the CLIP encoder. Given a mesh and style visual or textual guidance, the point diffusion model generates color for sampled points from the mesh, which are then projected onto 2D UV space to create a coarse texture image. Subsequently, the UV diffusion model utilizes the coarse texture and additional shape attributes to predict the high-fidelity texture.

3.3 Neural Field-Based Stylization

A neural field is "*a field that is parameterized fully or in part by a neural network*" (Po et al., 2023). The advanced 3D representations of neural fields, especially neural radiance fields (NeRFs) (Mildenhall et al., 2020; Sun et al., 2022; Müller et al., 2022; Kerbl et al., 2023), store scene geometry and appearance in a neural network or explicit data structure, enabling photorealistic rendering and 3D stylization in the latent space. Compared to mesh models (Fig. 2), which rely on texture maps to store various visual information like albedo, roughness, metalness, baked lighting, etc., neural fields store learned features that are mapped to an RGB image during rendering. Therefore, we can either stylize novel views during rendering without modifying the original neural field (Sect. 3.3.1), or stylize the neural field itself by updating the stored latent features (Sect. 3.3.2). Methods that stylize novel views learn a universal style transformation module for 3D-aware view features, thus avoiding additional training for each input style instance. While the other approaches that stylize neural fields require optimization for every single style or input reference set, the stylized neural field assets allow regular neural rendering, thus supporting seamless usage in related tools and software. Table 2 summarizes neural field-based stylization works, in terms of taxonomy and technical comparison. Please refer to the related surveys for a comprehensive review of neural fields and their applications (Xie et al., 2022; Chen & Wang, 2024b).

3.3.1 Feed-Forward Novel View Stylization

A straightforward approach to stylize a 3D scene is to stylize its novel views (Huang et al., 2021). However, it is known that a simple combination of existing 2D stylization and novel view synthesis methods would lead to blurry and inconsistent results. Instead, LSNV (Huang et al., 2021) proposed a feed-forward point cloud feature transformation model that first reconstructs a 3D point cloud by back-projecting the points in feature maps extracted from multi-view images following depth guidance, and then feeds these features into the transformation network (similar to LST (2019) in Sect. 2.1) to obtain stylized point cloud features, which can be decoded to render novel views.

Similarly to neural style transfer, one can also perform arbitrary style transfer for novel views of a NeRF scene (Chiang et al., 2022c; Huang et al., 2022; Liu et al., 2023; Kim et al., 2024). This involves a two-phase process consisting of geometry reconstruction training (NeRF training) and appearance stylization training. Chiang *et al*. (2022c) proposed to transfer arbitrary artistic style to novel views on NeRF++ representation (Zhang et al., 2020) for large outdoor 360° unbounded scenes. In the reconstruction phase, they separate geometry (density output) and appearance (view-dependent color output) into two branches, as illustrated in Fig. 9. In the stylization phase, they fix the geometry branch and use an MLP hypernetwork (Ha et al., 2016) with style features from a pre-trained VAE encoder to update the parameters of the appearance branch. Since NeRF++ cannot support high-resolution image rendering at a fast speed, they propose to do small-patch sub-sampling (Schwarz et

Table 2 Summary of selected neural field stylization methods. 3D Repr., Struct., geo., app. refer to 3D representation, data structure, geometry and appearance, respectively

Method	Base Technique	3D Repr.(Struct.)	Guidance	Scene	Optimization	Style Genre	⟨/⟩
LSNV (Huang et al., 2021)	FVS (Riegler & Koltun, 2020), LST (Li et al., 2019)	Point cloud	Image	Bounded/unbounded	Feed-forward	Appearance, artistic	
HyperStyle (Chiang et al., 2022c)	NeRF++ (Zhang et al., 2020), HyperNet (Ha et al., 2016)	NeRF	Image	Unbounded	Feed-forward	Appearance, artistic	
StyleRF (Liu et al., 2023)	TensoRF (Chen et al., 2022), SICT (Liu et al., 2023), LST (Li et al., 2019)	NeRF (grid)	Images	Bounded	Feed-forward	Appearance, artistic	
StylizedNeRF (Huang et al., 2022)	NeRF (Mildenhall et al., 2020), AdaIN (Huang & Belongie, 2017)	NeRF	Image	Bounded/unbounded	Feed-forward	Appearance, artistic	
UPST-NeRF (Chen et al., 2022b)	DVGO (Sun et al., 2022), I2I-GAN	NeRF (grid)	Image	Bounded	Feed-forward	Appearance, photoreal	
FPRF (Kim et al., 2024)	K-Planes (Fridovich-Keil et al., 2023), AdaIN (Huang & Belongie, 2017)	NeRF (grid)	Images	Unbounded	Feed-forward	Appearance, photoreal	
StyleGaussian (Liu et al., 2024)	3DGS (Kerbl et al., 2023), AdaIN (Huang & Belongie, 2017)	3DGS	Images	Bounded/unbounded	Feed-forward	Appearance	
ARF (Zhang et al., 2022)	Plenoxels (Fridovich-Keil et al., 2022), NNFM (Zhang et al., 2022)	NeRF (grid)	Image	Bounded/unbounded	Single-style	Appearance, artistic	
INS (Fan et al., 2022)	NeRF (Mildenhall et al., 2020), Gatys et al.(2016)	NeRF, SDF	Image	Bounded	Single/multi-style	Appearance, artistic	
SNeRF (Nguyen-Phuoc et al., 2022)	Any NeRF, any NST	NeRF	Image	Bounded/unbounded	Single-style	App.(&geo.), artistic	✗
LipNeRF (Zhang et al., 2023d)	Plenoxels (Fridovich-Keil et al., 2022)	NeRF (grid)	Image	Bounded/unbounded	Single-style	Appearance, photoreal	✗
Ref-NPR (Zhang et al., 2023c)	Plenoxels (Fridovich-Keil et al., 2022), any 2D edit	NeRF (grid)	Image(s)	Bounded/unbounded	Single-style	Appearance	
LocalStyleNeRF (Pang et al., 2023)	iNGP (Müller et al., 2022), NNFM (Zhang et al., 2022)	NeRF(hash grid)	Image	Room	Single/multi-style	Appearance, artistic	
SINE (Bao et al., 2023)	NeRF (Mildenhall et al., 2020), DIF (Deng et al., 2021), DINO (Caron et al., 2021)	NeRF	Image	Bounded/unbounded	Single-style	Geo&app	
NeRF-Art (Wang et al., 2023a)	VolSDF (Yariv et al., 2021), CLIP (Radford et al., 2021)	NeRF	Text	Bounded	Single-style	Appearance	
Instruct-N2N (Haque et al., 2023)	Nerfacto (Tancik et al., 2023), InstructPix2Pix (Brooks et al., 2023)	NeRF (hash grid)	Text	Bounded/unbounded	Single-style	Geo&app	
BlendingNeRF (Song et al., 2023)	NeRF (Mildenhall et al., 2020), CLIP (Radford et al., 2021)	NeRF	Text	Object	Single-style	Geo&app	✗
Vica-NeRF (Dong & Wang, 2023)	Nerfacto (Tancik et al., 2023), InstructPix2Pix (Brooks et al., 2023)	NeRF (hash grid)	Text	Bounded/unbounded	Single-style	Geo&app	
GaussianEditor (Chen et al., 2024d)	3DGS (Kerbl et al., 2023), any edit	3DGS	Text	Bounded/unbounded	Single-style	Geo&app	
GaussCtrl (Wu et al., 2024)	3DGS (Kerbl et al., 2023), any edit	3DGS	Text	Bounded/unbounded	Single-style	Geo&app	
GaussianGrouping (Ye et al., 2025)	3DGS (Kerbl et al., 2023), any edit	3DGS	Text	Bounded/unbounded	Single-style	Geo&app	
ReGS (Mei et al., 2024)	3DGS (Kerbl et al., 2023), any 2D edit	3DGS	Image(s)	Bounded/unbounded	Single-style	Appearance	✗

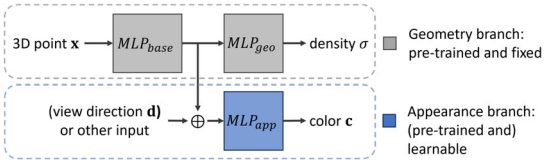


Fig. 9 Canonical NeRF with geometry and appearance branches in 3D stylization

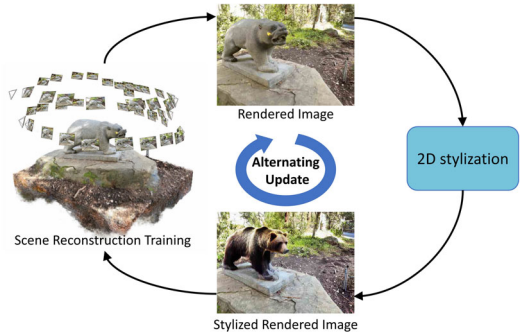


Fig. 10 A general framework for NeRF stylization through alternating updates of multi-views (Nguyen-Phuoc et al., 2022; Haque et al., 2023). Images adapted from Haque et al. (2023)

al., 2020) to compute content and style losses (Huang & Belongie, 2017).

As discussed in Sect. 2.1.2, AdaIN and LST are two mainstream techniques for arbitrary style transfer, which are adapted in StylizedNeRF (Huang et al., 2022) and StyleRF (Liu et al., 2023) respectively. StylizedNeRF applies a mutual learning strategy between a 2D arbitrary style transfer AdaIN (Huang & Belongie, 2017) model and a NeRF to achieve multi-view consistency and stylization of NeRF appearance. Specifically, they pre-train the style transfer model under the supervision of multi-view content, as well as style and additional consistency loss. During the mutual learning, they jointly train the style transfer model and a new appearance MLP branch with NeRF's fixed geometry branch.

StyleRF (Liu et al., 2023) employs a similar style transformation mechanism to LSNV, learning a 3D feature grid lifted from pre-trained VGG, and then applying linear style transformation on the weighted features of sampled points in the ray marching process of NeRF (Chen et al., 2022). Finally, a 2D CNN decoder is used to generate stylized views. StyleRF also conducts two-stage training, the stage of feature grid learning and reconstruction without viewing direction input (based on TensoRF (Chen et al., 2022)), and the stage of stylization training with fixed geometry. Moreover, StyleRF illustrated its advantage of data-driven style training for style interpolation and multi-style transfer with a 3D mask.

Later works further explore arbitrary photorealistic style transfer (Chen et al., 2022b), unbounded urban-scale scene style transfer (Kim et al., 2024) based on K-Planes representation and 2D-to-3D lifted DINO semantic features (Fridovich-Keil et al., 2023; Caron et al., 2021), and point cloud or mesh reconstruction of stylized novel views (Ibrahimli et al., 2024).

3.3.2 Optimization-Based Neural Field Stylization and Editing

In this section, we explore stylizing a neural radiance field (NeRF) by updating the scene information and features stored in neural networks or explicit data structure, rather than processing in the rendering phase. Most NeRF-based approaches for appearance optimization follow a two-step procedure (Zhang et al., 2022; Fan et al., 2022; Zhang et al., 2023c; Pang et al., 2023). Initially, the scene's geometry and appearance are reconstructed from multiple posed views before stylization. Subsequently, during the optimization of the appearance style, the geometry can be either fixed or self-distilled during stylization, as depicted in Fig. 9. Notably, some methods incorporate geometry updates during the stylization phase (Nguyen-Phuoc et al., 2022; Wang et al., 2023a; Haque et al., 2023).

A General Optimization Framework Nguyen-Phuoc et al. (2022) proposed SNeRF, a general alternating optimization pipeline for novel view stylization by arbitrary off-the-shelf 2D style transfer methods and any NeRF methods (Fig. 10). The proposed method follows a sequential process. Firstly, they reconstruct the NeRF scene with original content multi-views. Next, they iteratively stylize rendered multi-views and use stylized views to fine-tune the NeRF in a loop. Through several iterations, the entire NeRF representation gradually becomes 3D-aware stylized.

Haque et al. (2023) further extended the framework for text-guided NeRF editing and introduced Instruct-NeRF2NeRF, which edits a NeRF scene by leveraging 2D diffusion priors. Similar to SNeRF (2022), they adopted an iterative process to update training views and the NeRF scene alternatively (Fig. 10). They replace a training view by editing a rendered view from a training viewpoint using an off-the-shelf image-to-image diffusion model, e.g., Instruct-Pix2Pix (Brooks et al., 2023), then continue NeRF training on the updated training data.

Extensive experiments validated the flexibility of this general framework, showcasing its compatibility with off-the-shelf 2D style transfer methods across a range of NeRF or NeRF variants. Despite promising results, this framework is limited by its time-consuming iterative nature and is vulnerable to variations in stylized views, which can lead to style dilution and inconsistency. To address this issue, ViCA-NeRF (Dong & Wang, 2023) employs view-consistency-aware NeRF editing, which establishes explicit connections between different views and propagates the editing information from edited to unedited views.

Image-guided Radiance Field Stylization Image-guided neural style transfer (Sect. 2.1) has undergone significant development over the years and has inspired a multitude of works focusing on image-guided NeRF stylization. Similarly, most works follow a two-step optimization process, including NeRF reconstruction and appearance stylization stages.

Notably, ARF (Zhang et al., 2022) introduced the artistic radiance field approach, utilizing content loss (Gatys et al., 2016) and nearest neighbor feature matching (NNFM) style loss (Eq. 5) to optimize and stylize the appearance of a reconstructed NeRF scene with an exemplar style image. NNFM minimizes the cosine distance on VGG features between style reference and rendered image:

$$\mathcal{L}_{nnfm} = \frac{1}{N} \sum_{i,j} \min_{i',j'} D(F(cs)_{ij}, F(s)_{i'j'}), \tag{5}$$

where $F(\cdot)_{ij}$ denotes the feature vector at pixel location (i, j) of the feature map $F(\cdot)$. Experiments validated NNFM loss in NeRF stylization yields more visually appealing results than typical Gram matrix loss (Gatys et al., 2016) or CNNMRF loss (Li & Wand, 2016).

However, ARF still lacks explicit semantic correspondences. To address this limitation, Ref-NPR (Zhang et al., 2023c) proposed to first stylize a single view using structure-preserving 2D-stylization algorithms (Sect. 2.1) or manual editing, and then use this stylized reference view to construct a reference voxel dictionary for the scene, which enables matching of semantics and color features between edited view and the scene. The follow-up work CoARF (Zhang et al., 2024b) allows for style transfer with precise control over specific objects indicated by 2D segmentation masks. These semantics are also used for calculating the NNFM loss. Concurrently, ReGS (Mei et al., 2024) enables high-quality stylization that mimics reference textures while maintaining real-time rendering capabilities for free-view navigation. It achieves this by adapting 3D Gaussian Splatting (3DGS) and regularizing with scene depth.

Referring to the disentanglement of content and style representations for style transfer (Huang et al., 2018; Lee et al., 2018), Fan et al. (2022) proposed a generalizable model consisting of a style MLP and a content MLP to separately encode the style image and input scene, and an amalgamation MLP to output both final color and density, fusing style and content features. Additionally, Pang et al. (2023) considered semantic style matching and added additional segmentation output in the geometry branch with an extra segmentation MLP after the hash encoding process of iNGP (Müller et al., 2022). Both works proposed to use conditional style representations by feeding a one-hot vector or style index to the neural field, enabling conditional stylization for several styles (Fan et al., 2022; Pang et al., 2023).

Apart from the artistic style transfer approaches, LipRF (Zhang et al., 2023d) addressed the challenges in 3D photorealistic stylization by leveraging a Lipschitz MLP to transform the radiance appearance field during the stylization training stage. The scene views are first stylized by 2D photorealistic style transfer methods (Yoo et al., 2019; Wu et al., 2022) and then used to train the Lipschitz MLP.

Text-guided Radiance Field Stylization The recent increasingly developed vision-language models and text-to-image diffusion models (Sect. 2.1.3) inspire the community to develop works on text-guided or text-to-image guided 3D scene stylization and editing (Wang et al., 2022, 2023a, b; Song et al., 2023; Bao et al., 2023; Haque et al., 2023; Sella et al., 2023; Zhuang et al., 2023; Shum et al., 2024). Here we provide a brief discussion on some advances in text-guided NeRF stylization, showcasing their potential for rapid prototyping and customization of 3D asset designs.

NeRF-Art (Wang et al., 2023a) proposed text-guided NeRF stylization with profound semantics. Unlike simple color and shape stylization for objects in CLIP-NeRF (Wang et al., 2022) using a CLIP-based matching loss (Eq. 1), NeRF-Art realizes complex stylization on diverse shapes and scenes, such as turning a human face into a Tolkien elf. This method proposed to fine-tune the pre-trained NeRF (Yariv et al., 2021) using relatively direction CLIP loss (Eq. 2), local and global contrastive CLIP-based loss (Chen et al., 2020), perceptual loss (Johnson et al., 2016) and a weight regularization (Barron et al., 2022) for sharper details. The follow-up work Wang et al. (2023b) further enhances the semantics-aware stylization with a semantic contrastive loss and fine-tuned CLIP with ArtBench artwork database (Liao et al., 2022) for accurate artistic textual embedding.

Instead, SINE (Bao et al., 2023) employs a two-branch editing field to learn geometric and appearance adjustments. Given an edited view of a pre-trained NeRF scene, the method establishes the mesh prior using either DIF (Deng et al., 2021) for specific object categories or using ARAP (Sorkine & Alexa, 2007) with depth estimation (Bhat et al., 2021) and 2D feature matching (Jiang et al., 2021), and then composes the texture prior based on semantic features and structural self-similarity (Caron et al., 2021; Tumanyan et al., 2022). To preserve irrelevant areas, it distills a semantic feature field from DINO (Caron et al., 2021) and clusters features in the edited region of the edited view. To achieve precise manipulation of specific regions, Blending-NeRF (Song et al., 2023) leverages CLIPSeg (Lüddecke & Ecker, 2022), a pre-trained image segmentation model, and employs a region loss for supervision.

To support editing, DreamEditor (Zhuang et al., 2023) directly uses 2D diffusion priors to enable precise editing while keeping irrelevant regions untouched. Specifically, this method transforms the NeRF into a mesh-based neural field by marching cubes (Lorensen & Cline, 1987) and distillation,

with each mesh vertex assigned geometry and color features. To localize the edit regions aligned with a text prompt, the method leverages DreamBooth (Ruiz et al., 2023) to fine-tune Stable Diffusion (Rombach et al., 2022) using sampled views from a spherical viewing trajectory centered on the scene, and then retrieve 2D attention maps (Hertz et al., 2023b) as view masks that are later back-projected to 3D scene forming the 3D editing mask. Finally, geometry and color features, as well as mesh vertex positions in the 3D masked region are jointly optimized by the SDS loss (Sect. 2.2). Similar to implicit shape deformation methods (Bao et al., 2023; Gao et al., 2023), DreamEditor employs mesh vertex regularizers, including Laplacian rigidity losses (Sumner et al., 2007) among neighbor vertices, for smooth mesh deformation.

Facing the limitation of slow optimization in NeRF editing, ED-NeRF (Park et al., 2024) proposed to edit a latent NeRF using 2D latent diffusion priors (Rombach et al., 2022), improving editing efficiency. However, multi-view rendering of latent features lacks geometry consistency. Therefore, they introduce a novel refinement layer with ResNet blocks and self-attention layers to refine inconsistent multi-view latent features. During editing, ED-NeRF employs delta denoising score (DDS) (Hertz et al., 2023a) which is a difference between two SDS scores conditioned on two different text prompts, and a masked DDS for the target region segmented by CLIPSeg (Lüddecke & Ecker, 2022) and SAM (Kirillov et al., 2023) to keep irrelevant regions unchanged.

Recent advancements in 3DGS-based scene editing offer new solutions for efficient optimization, fine-grained control, and high-quality scene segmentation. GaussCtrl (Wu et al., 2024) and GaussianEditor (Chen et al., 2024d) introduce text-driven editing for 3D Gaussian Splatting. GaussCtrl emphasizes multi-view consistency and depth-conditioned editing, while GaussianEditor enhances control and precision using Gaussian semantic tracing and hierarchical splatting. Gaussian Grouping (Ye et al., 2025) extends Gaussian Splatting by incorporating identity encoding for object segmentation, enabling fine-grained scene understanding and versatile editing applications. These methods collectively enable real-time, high-quality, and efficient 3D scene manipulation across a wide range of applications, such as object removal, inpainting, and style transfer.

3.4 Volume Stylization

Compared to other representations, volume is an intuitive representation of 3D data, as naturally extended from 2D image representation. 3D neural stylization can be performed on a volume representation, e.g., image-guided neural style transfer on volumetric simulation, particularly dynamic smoke (Kim et al., 2019) and fluids (Kim et al., 2020). Table 3 summarizes works for volumetric stylization.

In Kim et al. (2019), they use pre-trained Inception CNN model (Szegedy et al., 2016) as the single-view feature extractor and apply content loss and style loss (Sect. 2.1) for semantics-aware abstract style transfer. They proposed a transport-based neural style transfer (TNST) method on grid-based voxels to optimize a velocity field (i.e., voxel movement) from several multi-views from Poisson sampling around a small area of the trajectory, and use a differentiable smoke renderer to render grayscale images to represent pixel-wise volume density. For temporal consistency among frames during smoke simulation, they compose a linear combination of the recursive aligned velocity fields of neighbor frames for the velocity field of the current frame. Later in Kim et al. (2020), they adopt particle-based attributes from mult-scale grids and optimize attributes of position, density, and color per particle, which intrinsically ensures better temporal consistency than recursive alignment of velocity fields (Kim et al., 2019). It largely improves efficiency by directly smoothing density gradients in stylization from adjacent frames for temporal consistency, and by stylizing only keyframes and interpolating particle attributes in between.

Subsequently, one can use a feed-forward network to achieve fast volumetric stylization (Aurand et al., 2022), reaching production-level quality (Kanyuk et al., 2023; Hoffman et al., 2023). One can also learn an arbitrary appearance style transfer model for volumetric simulation via a volume autoencoder (Guo et al., 2021).

3.5 Point Cloud Stylization

A point cloud is a discrete set of data points in 3D space, which may represent 3D shapes or objects. Each point can enclose additional attributes such as colors, normals (Pfister et al., 2000) and spherical harmonic coefficients (Kerbl et al., 2023) for rendering, or latent features (Huang et al., 2021) for 3D stylization.

There are a few works that attempted stylization for point clouds (Table 3). For example, PSNet (Cao et al., 2020) is a PointNet-based (Qi et al., 2017) stylization network for point cloud color and geometry style transfer with a point cloud example or an image example. Similar to representing content and style features from a pre-trained model (Gatys et al., 2016), PSNet uses a PointNet-based classifier with two separate shared MLPs to extract intermediate outputs as geometry/content representation and regard the Gram-matrix of these outputs as appearance/style representation. PSNet utilizes point-cloud-based content and style losses (Gatys et al., 2016) to optimize the geometry and/or appearance color of the source point cloud, simply replacing VGG features with PSNet features. Since the Gram-based style representation is invariant to the number or the order of the input points, an example style image treated as a set of points can

Table 3 Summary of selected neural stylization works for volume, point clouds and implicit shapes

Method	Input	Guidance	Output	Scene	Optimization	Style Genre	⟨/⟩
TNST (Kim et al., 2019)	Volume	Image	Volume	Dynamic smoke	Single-style	Geometry, artistic	✓
LNST (Kim et al., 2020)	Volume	Image	Volume	Fluid	Single-style	Geo&app., artistic	✓
SKPN (Guo et al., 2021)	Volume	Image	Volume	Dynamic/static model	Feed-forward	Appearance	✗
ENST (Aurand et al., 2022)	Volume	Image	Volume	Smoke	Multi-style, feed-forward	Geometry	✗
PSNet (Cao et al., 2020)	Colored point cloud	Image	Colored point cloud	Object	Single-style	Geo&app., photoreal	✓
PointInverter (Kim et al., 2023)	Point cloud	None	Point cloud	Object	Feed-forward	Geometry	✗
NeuralWavelet (Hu et al., 2024)	Implicit shape	None	Implicit shape	Object	Feed-forward	Geometry	✓
SPAGHETTI (Hertz et al., 2022)	Implicit shape	None	Implicit shape	Object	Feed-forward	Geometry	✓
SALAD (Koo et al., 2023)	Implicit shape	None, or text	Implicit shape	Object	Feed-forward	Geometry	✓

stylize source colored point cloud with only the target color style without shape deformation. For point clouds generated by a generative model, one can learn to map a point cloud to the latent space for editing. PointInverter (Kim et al., 2023) employed 3D point cloud GAN inversion and introduced an efficient way to conduct a 3D point cloud mapping to the latent space of a 3D GAN based on SP-GAN, a sphere-guided 3D point cloud generator (Li et al., 2021). PointInverter resolves the point ordering issue during 3D point cloud inversion, while preserving point correspondences, which enables point editing in latent space.

3.6 Implicit Shape Editing

An implicit primitive shape is a 3D surface represented by an implicit distance function, such as a signed distance function (SDF) or truncated signed distance function (TSDF). Powered by learning-based techniques, neural implicit shapes can represent 3D geometry as occupancy networks (Mescheder et al., 2019; Peng et al., 2020), distance fields (Park et al., 2019), volumetric radiance fields (Mildenhall et al., 2020), and Gaussian mixture models (GMMs). These neural continuous implicit representations have gained significant attention in the field of 3D shape generation and editing (Table 3).

Particularly, NeuralWavelet (Hu et al., 2024) utilized a compact wavelet representation consisting of coarse and detail coefficient volumes and designed a pair of diffusion generative models for coarse and detail 3D shape generation. During shape learning, an encoder is jointly trained to map the coarse coefficient volume to a condensed latent code. This latent code serves as a controllable condition for shape generation, inversion, and manipulation. Recent approaches such as SPAGHETTI (Hertz et al., 2022) and SALAD (Koo et al., 2023) adopted 3D generative models equipped with a hybrid representation that employs part-level disentanglement, extrinsic approximate shape and intrinsic geometric details disentanglement. In the hybrid representation, each part of the 3D shape is characterized by a set of extrinsic parameters, which form a Gaussian ellipsoid in 3D space (formulated by a 3D position with a covariance), capturing the approximate shape structure of that particular part. This part-level extrinsic-intrinsic disentanglement enables 3D shape generation and implicit shape manipulation such as local adjustment and part mixing. SALAD further incorporates text-guided shape part segmentation (Koo et al., 2022) and performs text-guided shape completion. These methods utilizing hybrid representations and incorporating text guidance offer promising advancements in 3D shape generation, editing, and manipulation, allowing for more intuitive and controlled transformations of 3D shapes.

3.7 Practical Guidelines

This section discusses practical aspects of 3D neural stylization methods, summarizing several design choices including 3D consistency, controllability, generalization, and efficiency.

3D Consistency. A particular challenge when performing stylization of 3D data is to ensure that view consistency is achieved so that the styles appear similar across views. We discuss common strategies to achieve view consistency, as follows.

- *View Sampling.* A reasonable camera sampling strategy is necessary for multi-view optimization without posed views. A common strategy is to (randomly) sample around predefined principal cameras or along the camera trajectory, as summarized in Table 4. Michel et al. (2022) devised a new training view selection scheme that samples view around an anchor view with the highest CLIP similarity with the target prompt. Data augmentation is a common trick as well, such as random perspective transformation and random resize plus crop (Michel et al., 2022; Ma et al., 2023; Chen et al., 2024e), rendering with random backgrounds (Hwang et al., 2023), mirroring and rotating subject elements (Aurand et al., 2022).
- *Constant Geometry.* Appearance-only 3D stylization requires keeping 3D geometry constant before and after stylization. The frequently used strategy is to fix geometry during the stylization stage. Particularly, some neural field stylization works use hyper MLPs to predict stylized appearance branch parameters for stylization while fixing geometry branch parameters for 3D geometric consistency (Chiang et al., 2022c; Chen et al., 2022b; Wang et al., 2023b).
- *View-independent Appearance.* Some works of appearance stylization tend to maintain multi-view color consistency. However, some 3D representations may lead to multi-view appearance inconsistency, for example, view-dependent effects in radiance fields. To preserve multi-view color consistency, scenes are often optimized without viewing direction input in radiance fields, sacrificing view-dependent effects for better multi-view appearance consistency (Zhang et al., 2022, 2023c; Liu et al., 2023; Pang et al., 2023).
- *2D Priors.* With the growing popularity of large-scale pre-trained vision models including VGG, CLIP, DINO, and diffusion models (Simonyan & Zisserman, 2015; Radford et al., 2021; Caron et al., 2021; Rombach et al., 2022; Zhang et al., 2023b), 3D-aware stylization can be achieved through multi-view optimization utilizing these 2D visual priors (Zhang et al., 2022; Wang et al., 2023a; Bao et al., 2023; Haque et al., 2023; Koo et al., 2024).
- *3D Priors.* With numerous well-collected 3D datasets, pre-trained point cloud priors (Qi et al., 2017; Li et al., 2021; Nichol et al., 2022) are popular to represent coarse geometry and facilitate geometric deformation among mesh, point cloud stylization works (Yin et al., 2021; Cao et al.,

Table 4 Summary of training view sampling in selected 3D neural stylization methods on object data. "Cam" refers to camera type such as orthogonal or perspective camera. "#View" is the sample number of views for each iteration/frame, unless indicated otherwise. "Aug" indicates rendered view augmentation

	Literature	Scene	Cam	#View	Aug	Sampling criteria
Mesh	Paparazzi (2018)	Object	Orth	1	✗	Uniform on offset surface, face vertex normal
	Text2Mesh (Michel et al., 2022)	Object	Pers	5[a]	✓	Text-guided anchor view, sample around anchor
	Text2Scene (Hwang et al., 2023)	Object	Pers	20[b]	✓	Random sample, evenly cover whole scene
	TEXTure (Richardson et al., 2023)	Object	Pers	8[b]	✗	Around object (incld. top/bottom)
	Text2Tex (Chen et al., 2023a)	Object	Pers	6+20[b,c]	✗	Axis-align, face surface normal
	StyleCity (Chen et al., 2024e)	City	Pers	1k+	✓	Look at several centroids, sample around scene
Volume	TNST (Kim et al., 2019)	Object	Orth	9	✗	Look at object, sample around a path
	SKPN (Guo et al., 2021)	Object	Pers	4	✗	Random sample in a path
	ENST (Aurand et al., 2022)	Object	Pers	1	✓	Sample in a path

[a] total views equals views per iteration;
[b] total views;
[c] 6 views for generation, 20 out of 36 predefined views for refinement

2020; Kim et al., 2023). Some other works leverage additional geometry guidance, such as depth map, normal map, and camera pose, for precise control of 3D-aware synthesis (Höllein et al., 2022; Yu et al., 2023; Guo et al., 2023).

• *Multi-view Attention.* Temporal attention mechanism is widely used in video generation methods (Blattmann et al., 2023), where an attention layer is applied to latent video frames to improve frame consistency. This concept can be adapted to the 3D domain. For example, VcEdit (Wang et al., 2025) inverse-renders the cross-attention maps from all views onto each Gaussian in the source 3DGS, creating an averaged 3D map. This 3D map is then rendered back to 2D, serving as the consolidated cross-attention maps for the originals, resulting in more coherent predictions.

Controllability. Stylization requires diversity and flexibility for users to design assets. We summarize some common strategies for different levels of control in 3D stylization.

• *Pre-trained Diffusion Models.* State-of-the-art diffusion models provide powerful controllability for content creation. For example, TextureDreamer (Yeh et al., 2024) uses DreamBooth (Ruiz et al., 2023) to distill texture information from input reference images, and ControlNet (Sect. 2.1.3, 2023b) to process additional 2D conditions, such as depth, normal, and edge maps.

• *Semantic Alignment.* Pre-trained segmentation models (e.g. Segment Anything (2023)), semantic labels or descriptions can be integrated to empower semantics-aware stylization. Table 5 shows some semantic matching tricks commonly used in 3D neural stylization. Some text-guided approaches rely on deliberate textual descriptions for local stylization (Michel et al., 2022; Ma et al., 2023; Wang et al., 2023b), while some approaches consider explicit visual semantic matching such as Text2Scene with 3D label inputs (Hwang et al., 2023). Gao et al. (2023) proposed a regularization term to preserve identity for smooth deformation. Some approaches consider using or lifting explicit semantic matching with off-the-shelf tools (Huang et al., 2022; Zhang et al., 2023c; Pang et al., 2023; Song et al., 2023; Kim et al., 2024). In addition, the reviewed works (Wang et al., 2023a, b) demonstrated that contrastive learning can effectively improve the learning of directional cues, such as textual semantics, in text-guided stylization.

• *Explicit Representation.* An explicit scene representation allows much easier control and more precise manipulation. For example, *DreamEditor* (Zhuang et al., 2023) transforms the NeRF into a mesh-based neural field with each mesh vertex assigned geometry and color features; Chen et al. (2024d); Hertz et al. (2022); Koo et al. (2023) use explicit 3D Gaussians for editing.

• *3D Shape Inversion.* A commonly adopted technique for shape manipulation is 3D shape inversion, which inverts 3D representation into latent space learned by large-scale pre-trained 3D generative models (Nichol et al., 2022; Qi et al., 2017). This approach enables shape editing in latent space and has been explored in various stylization works such as NeuralWavelet (2022), SPAGHETTI (2022), PointInverter (2023), and SALAD (2023).

Efficiency. Efficiency in 3D stylization is influenced by various factors such as style optimization methods and 3D representations. To enhance efficiency and minimize resource consumption, we present several tricks for efficiency improvement.

• *Optimize in Coarse-to-Fine Manner.* Instead of direct stylization in final high resolution, some works operate 3D style optimization in the low resolution and then apply decoding or super-resolution techniques for higher optimization efficiency. For example, Cao et al. (2023); Knodt and Gao (2023) optimize latent features for 3D consistency and effi-

Table 5 Summary of semantic alignment in selected 3D neural stylization methods

Literature	Semantic Matching Technique
Text2Mesh (Michel et al., 2022); X-Mesh (Ma et al., 2023)	Design a text prompt for part-level localization and stylization
TextDeformer (Gao et al., 2023)	Use the identity-preserving term to keep deformation not far from input mesh
Text2Scene (Hwang et al., 2023)	3D instance mesh label for instance-level stylization
TSNeRF (Wang et al., 2023b)	Design a text prompt for part-level localization and stylization
StyleRF (Liu et al., 2023)	Obtain 3D mask by NeRF-based object segmentation
Ref-NPR (Zhang et al., 2023c)	Semantic matched edited view as reference; Propagate style features to other views
LocalStyleNeRF (Pang et al., 2023)	Train NeRF to obtain 3D segmentation from multi-view segmentation maps
SINE (Bao et al., 2023)	Extract edited view's semantic features; Compute pixel-wise feature distance between edited region and training views
ED-NeRF (Park et al., 2024); BlendingNeRF (Song et al., 2023)	Off-the-shelf text-guided segmentation model to segment each view for semantic-aware training
DreamEditor (Zhuang et al., 2023)	2D attention maps in Diffusion as view masks, which are projected to 3D

cient diffusion. Guo et al. (2023); Wu et al. (2023) employ super-resolution models and Yu et al. (2023); Zeng et al. (2024) use texture refinement models to obtain high-quality outputs.

• *Scene Representations.* For neural fields, there are various advanced representations for fast training or rendering, such as Plenoxels (2022), SNeRG (2021), iNGP (2022), DVGO (2022), TensoRF (2022), MobileNeRF (2023c) and 3DGS (2024b). Table 2 features neural field stylization methods with applied base reconstruction techniques. Some 3D representations also tend to use advanced neural fields to represent style features such as neural style fields for meshes in Michel et al. (2022); Ma et al. (2023).

• *Optimize Rendering and Back-propagation.* In NeRF stylization, naive NeRF-based rendering is memory-intensive for a bulk of ray samplings and point queries, but style losses are based on full images. Therefore, some use patch-based training (Chiang et al., 2022c), deferred gradient back-propagation (Zhang et al., 2022), and some separate forward and back-forward steps with full-res computation and patch-wise back-propagation (Zhang et al., 2023d; Wang et al., 2023a, b).

• *Feed-forward Networks.* Instead of optimizing scene representation parameters, some works used feed-forward networks for efficient training/inference (Ma et al., 2023; Huang et al., 2021; Chen et al., 2024c; Aurand et al., 2022).

Generalization. Stylization techniques across different scenarios and datasets are crucial for real-world applications, such as gaming or entertainment industries. We discuss some designs for generalization here.

• *Universal Style Transfer Module.* Data-driven 3D neural stylization models train a universal 3D style transfer module that can generalize to new styles in a zero-shot manner. This type of work usually operates on novel view rendering rather than optimizing the scene features, as discussed in Sect. 3.3.1.

• *General Optimization Framework.* As presented in Sect. 3.3.2, SNeRF (Nguyen-Phuoc et al., 2022) and Instruct-NeRF2NeRF (Haque et al., 2023) introduced a general framework for a single scene optimization with either image or textual reference. They apply to any scene, any style, either geometry or appearance stylization.

4 Datasets and Evaluation

In this section, we summarize the frequently used datasets for 3D neural stylization, introduce existing evaluation metrics and criteria for 2D and 3D stylization, and provide a benchmark of state-of-the-art 3D neural stylization works as the reference for future works.

4.1 Datasets

Datasets are essential for effective training and thorough validating of 3D stylization models in terms of applicable scenarios, stylization diversity, etc. Table 6 illustrates selected popular 3D and 2D datasets for the evaluation of 3D neural stylization works, identifying their modality, scale, and other noteworthy features.

4.2 Criteria and Metrics

The stylization and evaluation of 3D assets are commonly conducted through multi-view renderings, which could be attributed to the inherent way how people perceive and process 3D stuff, and the advancement of 2D large pre-trained vision models. It is also possible to conduct stylization and

Table 6 Popular datasets for performance evaluation on 3D neural stylization. Pt. is the abbreviation of point cloud. [a]Train/Test sets

	Dataset Name	Content	#Scenes or Images	Synthetic/Real	Highlights	⟨/⟩
Mesh	TurboSquid (TurboSquid, 2023)	(textured) Mesh model	N/A	Synthetic	Professional publicly shared 3D models	
	CGTrader (CGTrader, 2023)	(textured) Mesh model	1.81M	Synthetic	Professional publicly shared 3D models	
	ShapeNet (Chang et al., 2015)	Mesh model	3M	Synthetic	Annotated 3D CAD object models	
	COSEG (Sidi et al., 2011)	Mesh model	172	Synthetic	Shapes with segment annotations	
	ModelNet (Wu et al., 2015)	Mesh model	151,128	Synthetic	3D CAD object models	
	Thingi10K (Zhou & Jacobson, 2016)	Mesh model	10K	Synthetic	3D-printing models	
	Objaverse 1.0/XL (Deitke et al. 2023a, 2023b)	(textured) Mesh model	800K/10M+	Synthetic	Annotated 3D objects and scenes sourced from Sketchfab	
Point Cloud	DensePoint (Cao & Nagao, 2019)	Colored point cloud	10,454	Synthetic	Point clouds built on ShapeNet and ShapeNetPart (Yi et al., 2016)	
View Synthesis	DTU (MVS) (Jensen et al., 2014)	multi-views, MVS	124	Real	Scanned objects	
	LLFF (Mildenhall et al., 2019)	Multi-views	24	Real	Forward-facing scenes	
	NeRF-Real (Mildenhall et al., 2020)	Multi-views	2	Real	Centered objects with background 360° Images	
	NeRF-Synthetic (Mildenhall et al., 2020)	Posed multi-views	8	Synthetic	Centered objects w/o background 360° images rendered by Blender	
	Tanks and Temples (Knapitsch et al., 2017)	Multi-views	21	Real	Large indoor and outdoor scenes	
	Real Lego (Fridovich-Keil et al., 2022)	Multi-views	1	Real	Centered Lego with background 360° images	
	Mip-NeRF 360 (Barron et al., 2022)	Multi-views	9	Real	Inward-facing object-centric 360° indoor and outdoor scenes	
Style	WikiArt (Painter by numbers, 2016)	Painting	79,433/23,815[a]	Real	Artistic paintings from WikiArt	
	DPST (Luan et al., 2017)	Photo	60	Real	Photorealistic stylish photos	
	COCO (Lin et al., 2014)	Photo	83K/41K[a]	Real	Large-scale objects in context	

evaluation directly in 3D space, mainly for the point cloud representation. For instance, 3DStyleNet (Yin et al., 2021) utilizes L1-Chamfer distance to guide the 3D shape transfer. PSNet (Cao et al., 2020) directly extracts style features from a point cloud using a modified PointNet (Qi et al., 2017) structure. SpiceE (Sella & Averbuch-Elor, 2023) introduces point cloud input as 3D shape prior to 3D generation. Achlioptas et al. (2023) provides a summary for evaluation metrics of 3D shape transfer.

We derive several critical aspects below from the state-of-the-art 3D neural stylization works for evaluating 3D stylization performance. Overall, the main consideration includes the alignment with the target style, the preservation of the original content, the consistency between different views, the visual quality of the stylized results, and the efficiency of training/inference.

• *Style Similarity* Stylization tasks are driven by guidance information (i.e. style reference), mainly images and texts. For measuring style similarity between the reference image and the rendering output, Gram matrix loss and AdaIN loss (i.e. MSE of the channel-wise mean and variance) that introduced in Sect. 2.1 are heavily used. When the style reference is given by textual prompt, the most popular choice of recent works is CLIPScore (Hessel et al., 2021), which quantifies the correspondence between the rendered image and the textual prompt.

• *Content Preservation* Content preservation is achieved to different extents in stylization works. In some works (Chiang et al., 2022c; Chen et al., 2022b; Wang et al., 2023b) the stylization is conducted only for appearance while the 3D geometry is locked, which dramatically eliminates the morphing of geometric content. Some other works (Zhang et al., 2022, 2023c) aim for a balance of stylization and content texture preservation by training exclusively with view-independent texture colors, hence showcasing multi-view color consistency.

• *Multi-view Consistency* Explicit 3D representations inherently provide multi-view geometry consistency. To measure multi-view appearance consistency, some 3D stylization works refer to video temporal short-range and long-range consistency evaluation (Lai et al., 2018), using warping difference error and the warped LPIPS, via optical flow estimation or depth estimation (Chiang et al., 2022c; Liu et al., 2023; Nguyen-Phuoc et al., 2022; Huang et al., 2021; Höllein et al., 2022). CLIP can also be applied to evaluate multi-view semantic consistency by encoding adjacent views to CLIP space as proposed in Haque et al. (2023); Ma et al. (2023).

• *Visual Quality* For image synthesis, we expect synthesized images to look natural and contain as few artifacts as possible. The inception score (IS)(Salimans et al., 2016) is designed to measure the image quality and diversity of generated images. Another popular metric is Frechet Inception Distance (FID) (Heusel et al., 2017), which compares the distribution of generated images with the distribution of real images. It works well to decide if generated images are similar to objects in the target domain. For instance, TSNeRF (Wang et al., 2023b) used FID to evaluate the distance between stylized rendered views and the target art database.

• *Robustness and Efficiency* For model robustness, Ref-NPR (Zhang et al., 2023c) proposed to measure PSNR between rendered results around a specific view angle. It is not essential for general 3D neural stylization evaluation, but it can be taken as a robustness reference. Regarding efficiency, important factors include the training time, inference speed, memory usage, data accessibility, model size and usability.

• *User Study* The above metrics do not necessarily reflect and align with human bias, especially for subjective factors such as naturalness and attractiveness. Therefore, conducting a user study is usually a suitable option. A typical user study involves steps including recruiting participants, preparing study materials and questionnaires, collecting answers, and analyzing statistics. In 3D stylization, the most frequently evaluated metrics are "stylization quality" and "temporal consistency" (Huang et al., 2021; Chiang et al., 2022c; Chen et al., 2022b; Liu et al., 2023).

4.3 Benchmark of 3D Stylization

In this section, we provide a benchmark evaluation in Table 7 of state-of-the-art mesh-based and neural field-based neural stylization methods in terms of the criteria discussed above, followed by a high-level discussion of the insights gained. The methods can be categorized into text-guided or image-guided object mesh texture stylization, text-guided neural field stylization, and image-guided neural field artistic stylization.

4.3.1 Experimental Settings

• *Datasets.* For mesh-based stylization methods with image guidance, we create 300 object-image pairs from Objaverse (Deitke et al., 2023) dataset: 100 objects with their own rendered images, 100 with rendered images of other objects in the same category, and 100 with rendered images from different categories. The first two parts aim to evaluate the capacity of "in-domain" texture transfer, while the last part tests the capacity of "out-of-domain" texture transfer. All the images are rendered in 2048 × 2048 resolution. For TEXTure (Richardson et al., 2023), we fine-tuned ten diffusion models following their official requirement to conduct the texture transfer. For mesh-based stylization methods with text guidance, we use the same 100 selected objects from Objaverse and directly use the object name to construct the text prompt.

For neural field-based stylization methods with image guidance, we include eight scenes from three public datasets,

Table 7 Evaluation of selected works across 3D representation and guidance type. w/depth stands for with pre-trained depth-ControlNet. SNeRF-G is reproduced by Plenoxels (2022) and Gatys et al. (2016)

3D Repr	Method	Guidance	Optimization	Style Similarity (ClipScore↑)	Multi-view Consistency (ClipVar↑)	Visual Quality (FID↓)	GPU (GiB)	Pre-training Time (min)	Optimization Time (min)
Mesh	TEXTure (Richardson et al., 2023)	Text	Single-style	**28.43**	81.39	201.77	∼ 11	No need	∼ 1.4
	Paint-it (Youwang et al., 2024)		Single-style	27.87	81.77	164.08	∼ 30	No need	∼ 30
	Paint3D (Zeng et al., 2024)		Single-style	27.54	80.41	167.13	∼ 25	No need	∼ 3
	TexPainter (Zhang et al., 2024d)		Single-style	26.73	82.62	157.92	∼ 14	No need	∼ 6
	FlashTex (w/depth)		Single-style	27.40	**84.36**	162.99	∼ 14	No need	∼ 2.9
	FlashTex (Deng et al., 2024)		Single-style	25.73	83.70	**155.73**	∼ 16	–	∼ 4.8

3D Repr	Method	Guidance	Optimization	Style similarity (ClipScore↑) in-domain/out-domain	Multi-view Consistency (ClipVar↑) in-domain/out-domain	Visual quality (FID↓)	GPU (GiB)	Pre-training Time (min)	Optimization Time (min)
Mesh	TEXTure (Richardson et al., 2023)	concrete object image	Single-style	79.70/64.66	86.07/**88.49**	172.44	∼ 11.5	∼ 140	∼ 0.9
	Paint3D (Zeng et al., 2024)		Single-style	**81.54**/70.23	**86.77**/85.51	133.01	∼ 26	No need	∼ 3.1
	Easi-Tex (Perla et al., 2024)		Single-style	81.37/**70.50**	84.75/83.50	**120.63**	∼ 12	No need	∼ 15.4

3D Repr	Method	Guidance	Optimization	Style Similarity (ClipScore↑)	Multi-view Consistency (ClipVar↑)	Visual Quality (FID↓)	GPU (GiB)	Pre-training Time (min)	Optimization Time (min)
Neural Field	InstructN2N (Haque et al., 2023)	text	Single-style	**26.47**	87.43	–	∼ 13	∼ 25	∼ 60
	VICA-NeRF (Dong & Wang, 2023)		Single-style	25.22	84.97	–	∼ 7.7	∼ 25	∼ 82
	InstructGS2GS (Vachha & Haque, 2024)		Single-style	22.16	86.26	–	∼ 4	∼ 5	∼ 18
	PDS (Koo et al., 2024)		Single-style	24.24	**87.63**	–	∼ 16	∼ 5	∼ 200

3D Repr	Method	Guidance	Optimization	Style Similarity (Gram Loss↓$\times 1e5$)	Multi-view Consistency (LPIPS↓$\times 1e2$) short-range/long-range	Visual Quality (FID↓)	GPU (GiB)	Pre-training Time (min)	Optimization Time (min)
Neural Field	ARF (Zhang et al., 2022)	artistic style image	Single-style	6.25	7.52/**4.60**	–	∼ 14	∼ 20	∼ 20
	SNeRF-G (Nguyen-Phuoc et al., 2022)		Single-style	**3.12**	11.72/4.62	–	∼ 14	∼ 20	∼ 100
	INS (Fan et al., 2022)		Single-style	4.78	11.15/8.12	–	∼ 24	∼ 2.3 × $1e3$	∼ 120
	Ref-NPR (Zhang et al., 2023c)		Single-style	4.47	8.02/5.30	–	∼ 14	∼ 20	∼ 20
	LSNV (Huang et al., 2021)		Feed-forward	9.09	8.84/8.05	**247.29**	∼ 12	∼ 1.7 × $1e3$	–
	HyperStyle (Chiang et al., 2022c)		Feed-forward	5.41	11.23/6.28	289.29	∼ 72	∼ 460	-
	StyleRF (Liu et al., 2023)		Feed-forward	4.84	6.12/5.42	267.11	∼ 24	∼ 300	–
	StyleGaussian (Liu et al., 2024)		Feed-forward	5.08	**5.51**/8.02	293.46	∼ 19	∼ 350	–

including single-object scenes (chair, mic) in NeRF-Synthetic (Mildenhall et al., 2020) that are inward-facing 360° objects without background, forward-facing real scenes (fern, flower, horns, trex) in LLFF dataset (Mildenhall et al., 2019), and large unbounded real scenes (Truck, Playground) in Tanks&Temples dataset (Knapitsch et al., 2017). Particularly, masked large scenes Caterpillar and Truck without background (Knapitsch et al., 2017) are used instead for StyleRF and INS. The style reference images are selected from WikiArt (Painter by numbers, 2016) dataset. 120 WikiArt (Painter by numbers, 2016) style references are used for feed-forward methods, and 6 WikiArt images for single-style optimization methods. We select artistic images here because neural field-based methods usually have larger scenes with multi-objects and backgrounds, and single-object images won't lead to satisfying results. Conversely, such artistic images with abstract concepts tend to ignore the detailed semantics of a concrete object, and thus are not suitable for most mesh stylization methods that focus on a single object. For neural field-based stylization methods with text guidance, we select two unbounded and two forward-facing scenes (farm, campsite; fangzhou, person) from InstructN2N (Haque et al., 2023) and test four style prompts for each of them.

• *Metrics.* For style similarity, we compute Gram Loss for artistic image-guided works and ClipScore for others. FID (Heusel et al., 2017) is adopted for measuring the visual quality of selected methods, using the rendered views of stylized results as evaluation samples and style reference images as ground truths. The original rendering views of selected objects from the Objaverse dataset are used as ground truths for evaluating text-guided mesh stylization works. Since the FID metric needs to be calculated on a large number of evaluation images, we skipped a few works that are hard to obtain a large amount of ground truth data (Haque et al., 2023; Dong & Wang, 2023; Vachha & Haque, 2024; Koo et al., 2024) or have a relatively slow optimization speed which prevents generating a sufficient number of evaluation samples (Fan et al., 2022; Nguyen-Phuoc et al., 2022; Zhang et al., 2022, 2023c).

For multi-view consistency, we utilize CLIP-Var (Li et al., 2025) to take the minimum value of cosine similarity between CLIP features of uniformly sampled views as a metric, which derives from the idea that images of the same object from multiple views have the same semantics. For the artistic style transfer task, we compute short-range and long-range warp error with masked LPIPS scores via off-the-shelf optical flow estimator RAFT (Teed & Deng, 2020).

The GPU consumption, pre-training time and optimization time are measured on RTX 5880 Ada GPUs with 48GB memory per GPU. The pre-training time denotes the normal duration for required additional training of the method before conducting stylization (while the original authors may have provided trained weights), like training a ControlNet (Deng et al., 2024), training a feature transformation module (Liu et al., 2023; Chiang et al., 2022c), training the original 3D reconstruction, etc. Please refer to our evaluation code repository for details: https://github.com/chenyingshu/advances_3d_neural_stylization.

4.3.2 Discussion

Through the theoretical analysis, benchmarking and practical experience, we aim to address a research question: *How do various factors such as 3D representation, optimization methods, guidance, etc., impact stylization outcomes across different dimensions like visual quality, consistency, and efficiency?* We will delve into this inquiry through the following key points.

• *Optimization—How to conduct efficient optimization?* When aiming for efficiency, effective strategies include employing large pre-trained models and training task-specific adapter modules. For example, TEXTure and FlashTex (Richardson et al., 2023; Deng et al., 2024) can synthesize stylish texture in high quality in under five minutes by leveraging large pre-trained diffusion models as priors. Additionally, some utilize feed-forward processing to enhance efficiency during stylization inference, removing the necessity for per-style optimization, as demonstrated in works like StyleRF and StyleGaussian(Liu et al., 2023, 2024).

• *Guidance—How to provide effective guidance?* In stylization, a visual prompt can efficiently convey intricate details, especially for complex designs or expectations that are hard to articulate in natural language. Conversely, textual prompts offer greater flexibility allowing for easy adjustments. In Table 7, image-guided mesh stylization methods (Richardson et al., 2023; Zeng et al., 2024; Perla et al., 2024) exhibit higher CLIP-scores compared to text-guided approaches (Richardson et al., 2023; Youwang et al., 2024; Zhang et al., 2024d; Deng et al., 2024). This disparity stems from the calculation principle of CLIP-score that captures multi-concept features from the inputs and measures the similarity of the features, where image-guided texture transfer can directly reconstruct the features from the reference image and thus easily achieve higher CLIP-scores. Meticulous prompt engineering is required to achieve similar results using natural language.

Beyond 2D guidance, 3D guidance proves effective for tasks like 3D shape transfer, often through the point cloud representation which enables 3D shape similarity calculation using metrics like Chamfer distance. The point cloud representation offers efficiency for physics simulation, scalability, and other advantages. 3D Gaussian Splatting (Kerbl et al., 2023) akin to point clouds has great potential in such topics (Kotovenko et al., 2024).

• *Visual Quality—How to enrich visual effects while reducing artifacts?* State-of-the-art 3D stylization methods improve visual quality by developing view-dependent appearances based on CG empirical models (Deng et al., 2024), utilizing multiple vision priors (Haque et al., 2023; Youwang et al., 2024; Zhang et al., 2022), and data-driven learning (Huang et al., 2021; Liu et al., 2023). For example, Easi-Tex (Perla et al., 2024) employs a pre-trained IP-Adapter (Ye et al., 2023) and an edge ControlNet to faithfully extract the texture and shape details respectively, demonstrating both high visual quality and style similarity, even if there are significant discrepancies between the input and the reference object (we denote it as "out-domain" type in Table 7). FlashTex (Deng et al., 2024) trains a novel LightControlNet, which learns from numerous rendered images of objects with different materials and enables providing rich visual details in texture generation.

• *Consistency - How to ensure multi-view consistency?* Existing works strive to achieve multi-view consistency when rendering photorealistic or artistic 3D scenes. As mentioned in the practical guidelines (Sect. 3.7), some works directly construct view-independent objects/scenes (Zhang et al., 2022; Liu et al., 2023; Zeng et al., 2024) which largely alleviates the worry. However, it will significantly improve the overall quality to provide view-dependent effects or make the object/scene light-aware (Zhang et al., 2024f; Deng et al., 2024). Similar to the ideas of linking 2D to 3D stylization in Sect. 2.1.4, one way is to devise a dedicated loss item to enforce multi-view consistency (Zhang et al., 2023c; Mei et al., 2024). ARF (Zhang et al., 2022) presents another way that applies a simple linear transformation of colors in RGB space for all rendered views to match the color statistics of the style image, which greatly improves the consistency between the rendered views. Last but not least, we can incorporate additional guidance (depth map, normal map, etc.) for generative models. As seen in Table 7, FlashTex (Deng et al., 2024) with depth ControlNet achieves the highest multi-view consistency among the text-guided mesh stylization methods. Compared to Paint3D and Easi-Tex which only use one image as style reference, TEXTure achieves an overall higher Clip variance, probably benefiting from fine-tuning a diffusion model with multi-view renderings of the target object for texture transfer.

• *Scalability - How to adapt stylization to different sizes of scenes?* Mesh-based stylization is thriving in 3D object assets, which is suitable for benchmarking with accessible 3D object datasets. Some researchers also attempt to stylize room-scale (Chen et al., 2024a; Höllein et al., 2022) or city-scale scenarios (Chen et al., 2024e) concerning complex semantics and view sampling strategies. By employing neural fields, 3D representations support various scales of scenes, from naive NeRF and grid-based radiance fields for object-centric and feed-forward scenes (Fan et al., 2022; Zhang et al., 2022; Liu et al., 2023) to 3DGS for scenes in the wild (Vachha & Haque, 2024; Liu et al., 2024). Stylization techniques can be tailored to specific representations considering the structure types such as grids (Liu et al., 2023), points (Huang et al., 2021), and optimization strategies, as summarized in Table 2 (Fig. 11).

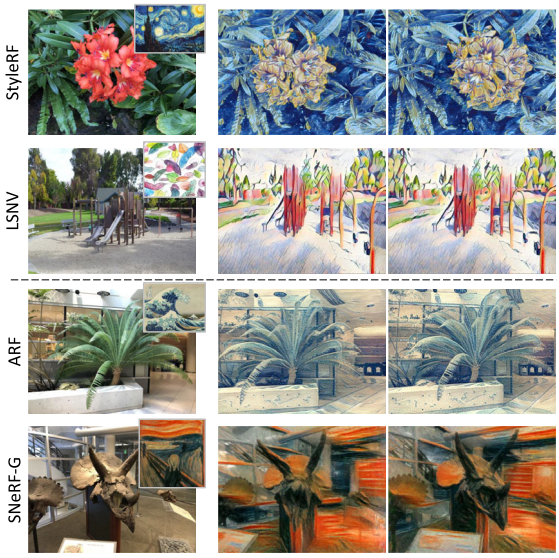


Fig. 11 Neural field stylization results. Zoom in for details

5 Applications

The burgeoning technologies for generating and manipulating 3D assets are unleashing the power of creativity and revolutionizing the way that we perceive and interact with visual content. 3D neural stylization sheds light on a new paradigm of providing infinite aesthetic possibilities from classic paintings to futuristic concepts, enhanced immersive experiences for virtual and augmented reality environments, seamless integration for cross-industry applications including advertising and marketing, fashion and product design, film and game development, architecture and environment visualization, and interactive education and learning, etc. Some examples are visualized in Fig. 12. In this section, we present some representative and promising applications of 3D neural stylization.

5.1 3D Asset Design

3D asset design and modeling involve constructing shapes, textures, materials, etc. Harnessing advanced neural stylization techniques, automatic 3D design becomes more flexible in a controllable way with text prompts, images or 3D references.

Fig. 12 Applications of 3D neural stylization. Images adapted from Orghidan et al. (2022); Chen et al. (2024a); Han et al. (2024); Volinga (2023); Kanyuk et al. (2023); Zhang et al. (2022); Wu et al. (2023); Li et al. (2023c)

5.1.1 Single Object Design

In the creation of consumer products, the stylization of a single object can enhance its market appeal. Neural stylization enables fast illustration of ideas for a more effective discussion among designers, developers and customers, especially in the prototyping phase. Text-guided mesh stylization provides a flexible way to design 3D assets. For example, after the launch of Text2Mesh (Michel et al., 2022), digital R&D designers and artists employed this tool for 3D gaming asset designs (Orghidan et al., 2022) and artwork creation (Guljajeva & Canet, 2022). Advanced techniques in automatic 3D shape mixing and morphing (Hui et al., 2022; Gao et al., 2023) are also promising for speedy 3D asset design and production.

5.1.2 Room Decoration

Current digital room/house decoration tools support adding pre-made assets to a virtual scene and adjusting their places, or using the device's camera to map the room and place provided furniture (Yu et al., 2011; Global, 2024; Houzz, 2024; Planner5D, 2024). However, the asset library provides limited types and styles of 3D models, and the overall style of the whole space is neglected. Recent works that leverage 3D neural stylization techniques explore more possibilities in room decoration. DreamSpace (Yang et al., 2024) allows users to personalize the appearance of real-world scene reconstructions with text prompts, and delivers immersive VR experiences on HMD devices. SceneTex (Chen et al., 2024a) generates high-quality textures for 3D indoor scenes from the given text prompts, which provide a consistent stylization for the whole space. Instead of entire room stylization, Text2Scene (Hwang et al., 2023) focuses on the stylization of individual object meshes in an indoor scene.

5.2 3D Avatar Stylization

Avatar stylization is a long-standing and popular research area, that enables interesting applications such as cartoonization for 2D or 3D-aware portraits (Jang et al., 2021; Song et al., 2021; Yang et al., 2022; Zhang et al., 2023a). With neural stylization techniques and novel 3D representations such as NeRF, there appear stylization solutions for 3D avatars (Pérez et al., 2024; Zhang et al., 2024c; Han et al., 2024).

For example, the general NeRF stylization framework SNeRF (Nguyen-Phuoc et al., 2022) supports style transfer for dynamic NeRF avatars. 3DFaceHybrid (Feng & Singhal, 2024) achieves arbitrary style transfer for a NeRF-based face by lifting 2D pre-trained face style transfer knowledge (Yang et al., 2022) to the 3D face mesh. StyleAvatar (Pérez et al., 2024) enables either image- or text-guided stylization for animatable avatars from a phone scan (Cao et al., 2022) with CLIP supervision. TECA (Zhang et al., 2024c) generates a detailed 3D avatar composition based on a given text description. The avatar includes a mesh-based face and body, and NeRF-based hair, clothing, and other accessories. HeadSculpt (Han et al., 2024) generates and edits a 3D-consistent head avatar with text prompts via diffusion priors (Brooks et al., 2023), achieving editions such as realistic or artistic head generation, expression editing, cartoonization, etc.

5.3 Non-Photorealistic Rendering

Compared to traditional non-photorealistic rendering techniques (Gooch et al., 1998; Gooch & Gooch, 2001) with low-level control of simple strokes and textures, neural stylization techniques realize general stylization for arbitrary style targets, offering high-level controllability with reference and semantics and higher speed for stylized assets production. 3D artistic stylization works have shown the potential of efficient NPR for a scene (Huang et al., 2021; Chiang et al., 2022c; Huang et al., 2022; Liu et al., 2023; Zhang et al., 2022; Fan et al., 2022; Nguyen-Phuoc et al., 2022; Zhang et al., 2023c; Wang et al., 2023a; Haque et al., 2023).

5.4 Physically Based Rendering

Physical properties in 3D scenes enable photorealistic rendering and editing.

5.4.1 Texture Stylization

TANGO (Lei et al., 2022) tends to optimize texture material parameters by CLIP supervision. It trains MLPs given the point and its normal to generate SVBRDF parameters and normal offset, which enables photorealistic rendering. Its follow-up work 3DStyle-Diffusion (Yang et al., 2023) further incorporates depth-guided ControlNet (Zhang et al., 2023b) for score distillation, enabling high-quality fine-grained texture stylization.

5.4.2 3D Generation and Editing

HyperDreamer (Wu et al., 2023) achieves single-image-to-3D generation with physical decomposition including semantics, albedo, specular, roughness, and normal. It supports diverse downstream tasks such as relighting, text-guided and part-aware editing. Decorate3D (Guo et al., 2023) converts NeRF scene to mesh for geometry and material decomposition. The decoupled geometry and UV texture representations support controllable texture editing and generation with text instructions.

5.4.3 Simulation

ClimateNeRF (Li et al., 2023c) fuses weather physical simulation with NeRF rendering to create NeRF scenes with realistic weather effects such as smog, snow, and floods. PhysGaussian (Xie et al., 2024) integrates physics-based dynamics simulation, specifically the Material Point Method (MPM) simulation, to deform a 3DGS scene. By merging realistic rendering and physical simulation, these approaches have the potential to enhance the realism of virtual games and films.

5.5 Industrial Production

3D neural stylization provides automatic stylization techniques for 3D assets including mesh, point cloud, volumetric simulation, and novel views. Stylized assets can be seamlessly integrated into traditional computer graphics rendering pipelines and software, such as meshes with new stylized texture, re-colored point clouds, and stylized volumetric simulation. Implicit reconstructed scenes, such as NeRF, can be exported as textured mesh or rendered by game engine plugins such as Luma AI's Unreal Engine NeRF plug-ins (Luma, 2023). Automated 3D environment synthesis holds great promise for film **virtual production** applications. For instance, combining environmental NeRF with light stages (Manzaneque, 2023) enables cost-effective scene shooting using Volinga suite (Volinga, 2023). Non-photorealistic stylized 3D assets and scenes are particularly beneficial for **animation production**, as demonstrated by the film Elemental (Hoffman et al., 2023; Kanyuk et al., 2023). Moreover, these techniques find possible applications in VR and video game development (Menapace et al., 2022), enabling rapid stylization and editing of 3D scenes (Liu et al., 2023; Fang et al., 2023).

6 Open Challenges and Future Works

From this survey, we identify under-explored problems and notable challenges in 3D neural stylization that are worth investigating in future work, which we discuss below.

6.1 Generalization

6.1.1 Large-scale Scene Stylization

Most 3D neural stylization works focus on objects or object-centric scenes (Michel et al., 2022; Hertz et al., 2022), room-scale scenes (Pang et al., 2023; Höllein et al., 2022), and outdoor inward-facing scenes (Huang et al., 2021; Chiang et al., 2022c). Though G. Kim et al. (2024) extended novel view stylization to city scenes, it does not output stylized 3D assets. StyleCity (2024e) stylizes urban texture and sky but replies on heavy mesh representation. 3D assets and scenes can scale to as large as multi-room indoor scene (Straub et al., 2019; Huang et al., 2022), architectural scenes (Martin-Brualla et al., 2021; Wang et al., 2021), multi-block outdoor scenes (Tancik et al., 2022; Turki et al., 2023), and even city-scale scenes (Xiangli et al., 2022; Xu et al., 2023; Li et al., 2023b). These complex scenarios with intricate semantics are challenging for semantic alignment and computation efficiency.

6.1.2 4D Scene Stylization

Limited literature exists on 4D scene stylization, with only a few notable works such as SNeRF (2022) and S-DyRF (2024b) presenting to stylize dynamic portraits and small scenes. Stylizing time-varying scenes with dynamic geometry and appearance changes (Park et al., 2021; Song et al., 2023; Yang et al., 2024), or integrating time-related special effects (Shih et al., 2013; Logacheva et al., 2020), poses significant challenges in maintaining temporospatial consistency in this domain.

6.1.3 Generalizable Text-guided Stylization

Text-guided 3D scene stylization and editing are still in the early stages. Data-driven generalizable text-guided 3D scene stylization or editing *without re-training* is seldom explored yet (Fang et al., 2023), which demands more attention. It is worth investigation and exploration since current advanced large-language models such as BLIP (Li et al., 2023a), GPT (Brown et al., 2020) enable infinite image-text pair data generation for data-driven model training.

6.2 Controllability

6.2.1 3D Reference-Guided Stylization

Various modalities have been explored to be style references, especially images and text prompts. 3D to 3D geometric and appearance style transfer with 3D shape or 3D scene guidance is still underexplored (Yin et al., 2021). While 3D features can provide 3D-aligned holistic references for stylization, 3D feature extraction suffers from limited 3D datasets in a few categories. With the rapid development of 2D-to-3D lifting techniques, there is potential to leverage large-scale pre-trained 2D models, such as 2D diffusion models (Rombach et al., 2022; Zhang et al., 2023b), as priors for context-aware scene stylization with 3D references. In addition, we expect more and more 3D pre-trained feature extractors and generative models with data in the wild to boost the 3D context-aligned style transfer.

6.2.2 Multi-modal Controls

Currently, most research works focus on sole-reference guidance for stylization, while multi-modal reference can provide high accuracy and controllability on precise manipulation and design (Pang et al., 2023; Bao et al., 2023; Zhuang et al., 2024). Therefore, it is worthwhile to explore the possibilities of joint supervision incorporating visual (Simonyan & Zisserman, 2015), textual (Radford et al., 2021), semantic (Caron et al., 2021), and geometric features for 3D stylization.

6.3 Efficiency

6.3.1 Real-time Arbitrary Style Transfer of 3D Scenes

Modern photo or video filters support real-time processing (Ruder et al., 2016; Jamriška et al., 2019). For instance, Ioannou and Maddock (2023) proposed a simplified style transfer architecture embedded into Unity rendering pipeline, enabling real-time depth-aware 2D style transfer. However, real-time stylizing a 3D scene given arbitrary styles remains challenging for some 3D representations due to the slow optimization process (Michel et al., 2022; Richardson et al., 2023; Cao et al., 2023; Yang et al., 2023) and slow rendering in neural fields (Chiang et al., 2022c; Zhang et al., 2022). Even though there are some novel view stylization works (Li et al., 2019; Liu et al., 2023; Chen et al., 2024c) that achieve arbitrary style transfer for speedy novel view synthesis, they fail to obtain stylized 3D scenes instantly. It is worth exploring to improve stylization speed by leveraging feed-forward networks (Aurand et al., 2022), 3D generative models (Cao et al., 2020), and advanced 3D representations such as 3DGS (Liu et al., 2024; Zhang et al., 2024a).

6.4 3D Consistency

6.4.1 Comprehensive View Planning for Complex Scenes

Existing works have primarily focused on planning training views for object or room scenes for 3D stylization (Michel et al., 2022; Hwang et al., 2023; Richardson et al., 2023; Chen et al., 2023a), overlooking the crucial aspects of semantic- and instance-level view planning. Hence, it is a compelling research opportunity to investigate effective strategies for planning views in scenes characterized by intricate semantics such as cityscapes and multi-room scenarios.

6.4.2 3D-Holistic Style Feature of Scenes

The majority of works reviewed are supervised by large-data 2D pixel-level features extracted from multi-views (Michel et al., 2022; Zhang et al., 2022; Kim et al., 2019), since large-data 3D pre-trained models are still rare and expensive. Even though some works try to lift 2D content features to 3D before 3D stylization (Huang et al., 2021; Liu et al., 2023; Huang et al., 2022), they still use view-dependent style features for final 3D stylization supervision. It is also impractical to lift 2D features to 3D at every iteration. Some works supervise stylization with a 3D-aware style feature by averaging features of several views for a small object (Michel et al., 2022; Ma et al., 2023), which is not implementable for more views with limited memory. Per-view or multi-view supervision may not be sufficient to represent the whole 3D scene style feature, and worse may dilute the current single-view style from other views with conflicting gradients (Gao et al., 2023). More research and investigation are needed for efficient 3D-aware and even 3D-holistic style features for 3D stylization.

6.5 Evaluation

6.5.1 Standardized Evaluation Across Modalities

The current evaluation metrics do not always align with human preference. User study is still widely adopted but it also prevents a precise quantitative analysis of method perfor-

mance. The heterogeneity of datasets of different modalities also imposes great challenges for a fair and comprehensive comparison of works on different modalities. We believe there should be some variations from the evaluation of our benchmark, while concerns should be similar to criteria in Sect. 4.2.

7 Conclusion

The report has explored the advancements in neural stylization techniques for diverse 3D data, including mesh, volume, neural fields, point cloud, and implicit shapes. Through this comprehensive survey of 3D neural stylization techniques and corresponding applications, we highlighted the importance of neural stylization in accelerating the creative process, enabling fine-grained control over stylization, and enhancing artistic expression in various domains such as movie making, virtual production, and video game development. Furthermore, we have introduced a taxonomy for neural stylization, providing a framework for categorizing new works in the neural stylization field. Our analysis and discussion of advanced techniques underscored the ongoing research efforts aimed at addressing limitations and pushing the boundaries of neural stylization in the 3D digital domain. In addition, we proposed a benchmark of 3D neural stylization, with which we aim to offer reference and inspiration for future 3D stylization works. Finally, we introduced practical applications and discussed open challenges and future works of 3D neural stylization.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11263-025-02403-9.

Acknowledgements The paper was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. HKUST 16202323) and an internal grant from HKUST (R9429). Binh-Son Hua is supported by Research Ireland under the Research Ireland Frontiers for the Future Programme—Project, award number 22/FFP-P/11522.

Funding Open access funding provided by Hong Kong University of Science and Technology. The paper was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. HKUST 16202323) and an internal grant from HKUST (R9429). Binh-Son Hua is supported by Research Ireland under the Research Ireland Frontiers for the Future Programme—Project, award number 22/FFP-P/11522.

Data Availability The experimental data for the benchmark of 3D stylization are available within the paper (Table 6). The evaluation codes are available via the open repository: https://github.com/chenyingshu/advances_3d_neural_stylization.

References

Achlioptas, P., Huang, I., Sung, M., Tulyakov, S., & Guibas, L. (2023). Shapetalk: A language dataset and framework for 3d shape edits and deformations. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 12685–12694).

Aigerman, N., Gupta, K., Kim, V. G., Chaudhuri, S., Saito, J., & Groueix, T. (2022). Neural jacobian fields: Learning intrinsic mappings of arbitrary meshes. *ACM Transactions on Graphics (TOG), 41*(4), 1–17. https://doi.org/10.1145/3528223.3530141

Artineering (2018). *MNPR.* https://artineering.io/software/maya-npr

Aurand, J., Ortiz, R., Nauer, S., & Azevedo, V. C. (2022). Efficient neural style transfer for volumetric simulations. *ACM Transactions on Graphics (TOG), 41*(6), 1–10. https://doi.org/10.1145/3550454.3555517

Avrahami, O., Lischinski, D., & Fried, O. (2022). Blended diffusion for text-driven editing of natural images. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 18208–18218).

Baatz, H., Granskog, J., Papas, M., Rousselle, F., & Novák, J. (2022). Nerf-tex: Neural reflectance field textures. *Computer Graphics Forum, 41*, 287–301.

Bao, C., Zhang, Y., Yang, B., Fan, T., Yang, Z., Bao, H., & Cui, Z. (2023). Sine: Semantic-driven image-based nerf editing with prior-guided editing field. In *The ieee/cvf computer vision and pattern recognition conference (cvpr).*

Barron, J., Mildenhall, B., Verbin, D., Srinivasan, P.P., & Hedman, P. (2022). Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 5470–5479).

Bar-Tal, O., Yariv, L., Lipman, Y., & Dekel, T. (2023). Multidiffusion: Fusing diffusion paths for controlled image generation. In *Proceedings of the 40th international conference on machine learning.*

Bhat, S., Alhashim, I., & Wonka, P. (2021). Adabins: Depth estimation using adaptive bins. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 4009–4018).

Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., & others (2023). Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv:2311.15127

Brooks, T., Holynski, A., & Efros, A.A. (2023). Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 18392–18402).

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., & others (2020). Language models are few-shot learners. In *Advances in neural information processing systems* (Vol. 33, pp. 1877–1901).

Cao, C., Simon, T., Kim, J. K., Schwartz, G., Zollhoefer, M., Saito, S.-S., et al. (2022). Authentic volumetric avatars from a phone scan. *ACM Transactions on Graphics (TOG), 41*(4), 1–19. https://doi.org/10.1145/3528223.3530143

Cao, T., Kreis, K., Fidler, S., Sharp, N., & Yin, K. (2023). Texfusion: Synthesizing 3d textures with text-guided image diffusion models.

In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 4169–4181).

Cao, X., & Nagao, K. (2019). Point cloud colorization based on densely annotated 3d shape dataset. In *Multimedia modeling: 25th international conference, mmm 2019, thessaloniki, greece, january 8–11, 2019, proceedings, part i 25* (pp. 436–446).

Cao, X., Wang, W., Nagao, K., & Nakamura, R. (2020). Psnet: A style transfer network for point cloud stylization on geometry and color. In *Proceedings of the ieee/cvf winter conference on applications of computer vision* (pp. 3337–3345).

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *Proceedings of the international conference on computer vision (iccv)*.

CGTrader (2023). *Cgtrader.* https://www.cgtrader.com/

Chan, E., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., De Mello, S., & others (2022). Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 16123–16133).

Chang, A., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., & Yu, F. (2015). *ShapeNet: An Information-Rich 3D Model Repository* (Tech. Rep. No. arXiv:1512.03012 [cs.GR]). Stanford University - Princeton University - Toyota Technological Institute at Chicago.

Chang, H., Wang, Z., & Chuang, Y.-Y. (2020). Domain-specific mappings for generative adversarial style transfer. In *European conference on computer vision* (pp. 573–589).

Chen, A., Xu, Z., Geiger, A., Yu, J., & Su, H. (2022). Tensorf: Tensorial radiance fields. In *European conference on computer vision* (pp. 333–350).

Chen, D., Li, H., Lee, H.-Y., Tulyakov, S., & Nießner, M. (2024). Scenetex: High-quality texture synthesis for indoor scenes via diffusion priors. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Chen, D., Siddiqui, Y., Lee, H.-Y., Tulyakov, S., & Nießner, M. (2023). Text2tex: Text-driven texture synthesis via diffusion models. In *roceedings of the ieee/cvf international conference on computer vision (iccv).*

Chen, G., & Wang, W. (2024). A survey on 3d gaussian splatting. arXiv:2401.03890

Chen, J., Xing, W., Sun, J., Chu, T., Huang, Y., Ji, B., & Wang, Z. (2024). Pnesm: Arbitrary 3d scene stylization via prompt-based neural style mapping. *Aaai.*

Chen, R., Chen, Y., Jiao, N., & Jia, K. (2023). Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the ieee/cvf international conference on computer vision (iccv).*

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597–1607).

Chen, Y., Chen, Z., Zhang, C., Wang, F., Yang, X., Wang, Y., & Lin, G. (2024). Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Chen, Y., Huang, H., Vu, T.-A., Shum, K.C., & Yeung, S.-K. (2024). Stylecity: Large-scale 3d urban scenes stylization. In *European conference on computer vision* (pp. 1–18).

Chen, Y., Vu, T.-A., Shum, K.-C., Hua, B.-S., & Yeung, S.-K. (2022). Time-of-day neural style transfer for architectural photographs. In *2022 ieee international conference on computational photography (iccp)* (pp. 1–12).

Chen, Y., Yuan, Q., Li, Z., Liu, Y., Wang, W., Xie, C., & Yu, Q. (2022). Upst-nerf: Universal photorealistic style transfer of neural radiance fields for 3d scene. *arxiv.*

Chen, Z., Funkhouser, T., Hedman, P., & Tagliasacchi, A. (2023). Mobilenerf: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 16569–16578).

Chen, Z., & Zhang, H. (2019). Learning implicit fields for generative shape modeling. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 5939–5948).

Chiang, P., Tsai, M.-S., Tseng, H.-Y., Lai, W.-S., & Chiu, W.-C. (2022). Stylizing 3d scene via implicit representation and hypernetwork. In *Proceedings of the ieee/cvf winter conference on applications of computer vision* (pp. 1475–1484).

Chung, J., Hyun, S., & Heo, J.-P. (2024). Style injection in diffusion: A training-free approach for adapting large-scale diffusion models for style transfer. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 8795–8805).

Croitoru, F., Hondru, V., Ionescu, R.T., & Shah, M. (2023). Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.,

Deitke, M., Liu, R., Wallingford, M., Ngo, H., Michel, O., Kusupati, A., & Farhadi, A. (2023). Objaverse-XL: A universe of 10m+ 3d objects. *Thirty-seventh conference on neural information processing systems datasets and benchmarks track.* https://openreview.net/forum?id=Sq3CLKJeiz

Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., & Farhadi, A. (2023). Objaverse: A universe of annotated 3d objects. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 13142–13153).

Deng, K., Omernick, T., Weiss, A., Ramanan, D., Zhu, J.-Y., Zhou, T., & Agrawala, M. (2024). Flashtex: Fast relightable mesh texturing with lightcontrolnet. In *European conference on computer vision.*

Deng, Y., Tang, F., Dong, W., Ma, C., Pan, X., Wang, L., & Xu, C. (2022). Stytr2: Image style transfer with transformers. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 11326–11336).

Deng, Y., Yang, J., & Tong, X. (2021). Deformed implicit field: Modeling 3d shapes with learned dense correspondence. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 10286–10296).

Dong, J., & Wang, & Y.-X. (2023). Vica-nerf: View-consistency-aware 3d editing of neural radiance fields. In *Thirty-seventh conference on neural information processing systems.*

Dumoulin, V., Shlens, J., & Kudlur, M. (2017). A learned representation for artistic style. In *International conference on learning representations.*

Efros, A.A., & Freeman, W.T. (2001). Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on computer graphics and interactive techniques* (pp. 341–346).

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., & others (2024). Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning.*

Everaert, M.N., Bocchio, M., Arpa, S., Süsstrunk, S., & Achanta, R. (2023). Diffusion in style. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 2251–2261).

Fan, Z., Jiang, Y., Wang, P., Gong, X., Xu, D., & Wang, Z. (2022). Unified implicit neural stylization. In *European conference on computer vision* (pp. 636–654).

Fang, S., Wang, Y., Yang, Y., Tsai, Y.-H., Ding, W., Zhou, S., & Yang, M.-H. (2023). Editing 3d scenes via text prompts without retraining. https://doi.org/10.48550/arXiv.2309.04917

Feng, J., & Singhal, P. (2024). 3d face style transfer with a hybrid solution of nerf and mesh rasterization. In *Proceedings of the ieee/cvf winter conference on applications of computer vision* (pp. 3504–3513).

Frenkel, Y., Vinker, Y., Shamir, A., & Cohen-Or, D. (2024). Implicit style-content separation using b-lora. arXiv:2403.14572

Fridovich-Keil S. and YU A., Tancik, M., Chen, Q., Recht, B., & Kanazawa, A. (2022). Plenoxels: Radiance fields without neural networks. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Fridovich-Keil, S., Meanti, G., Warburg, F.R., Recht, B., & Kanazawa, A. (2023). K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 12479–12488).

Fuji Tsang, C., Shugrina, M., Lafleche, J.F., Takikawa, T., Wang, J., Loop, C., & Lebaredian, R. (2022). *Kaolin: A pytorch library for accelerating 3d deep learning research.*https://github.com/NVIDIAGameWorks/kaolin.

Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A.H., Chechik, G., & Cohen-Or, D. (2022). An image is worth one word: Personalizing text-to-image generation using textual inversion. arXiv:2208.01618

Gal, R., Patashnik, O., Maron, H., Bermano, A. H., Chechik, G., & Cohen-Or, D. (2022). Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG), 41*(4), 1–13. https://doi.org/10.1145/3528223.3530164

Gao, W., Aigerman, N., Groueix, T., Kim, V., & Hanocka, R. (2023). Textdeformer: Geometry manipulation using text guidance. In *Acm siggraph 2023 conference proceedings* (pp. 1–11).

Gatys, L., Ecker, A., & Bethge, M. (2016). Image style transfer using convolutional neural networks. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 2414–2423).

Global, I. (2024). *Ikea place.* https://www.ikea.com/

Gooch, A., Gooch, B., Shirley, P., & Cohen, E. (1998). A non-photorealistic lighting model for automatic technical illustration. In *Proceedings of the 25th annual conference on computer graphics and interactive techniques* (pp. 447–452).

Gooch, B., & Gooch, A. (2001). *Non-photorealistic rendering*. AK Peters/CRC Press.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., & Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems* (Vol. 27).

Gu, J., Liu, L., Wang, P., & Theobalt, C. (2022). Stylenerf: A style-based 3d aware generator for high-resolution image synthesis. In *International conference on learning representations.* https://openreview.net/forum?id=iUuzzTMUw9K

Guljajeva, V., & Canet, M. (2022). *Psychedelic forms.* https://var-mar.info/psychedelic-forms/

Guo, J., Li, M., Zong, Z., Liu, Y., He, J., Guo, Y., & Yan, L.-Q. (2021). Volumetric appearance stylization with stylizing kernel prediction network. *ACM Transactions on Graphics (TOG), 40*(4), 162–1. https://doi.org/10.1145/3450626.3459799

Guo, Y., Zuo, X., Dai, P., Lu, J., Wu, X., Cheng, L., & Wu, X. (2023). Decorate3d: Text-driven high-quality texture generation for mesh decoration in the wild. In *Thirty-seventh conference on neural information processing systems.*

Ha, D., Dai, A., & Le, Q.V. (2016). Hypernetworks. https://doi.org/10.48550/arXiv.1609.09106

Haeberli, P. (1990). Paint by numbers: Abstract image representations. In *Proceedings of the 17th annual conference on computer graphics and interactive techniques* (pp. 207–214).

Haetinger, G.G., Tang, J., Ortiz, R., Kanyuk, P., & Azevedo, V. (2024). Controllable neural style transfer for dynamic meshes. In *Acm siggraph 2024 conference papers.* New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3641519.3657474

Han, X., Cao, Y., Han, K., Zhu, X., Deng, J., Song, Y.-Z., & Wong, K.-Y.K. (2024). Headsculpt: Crafting 3d head avatars with text. *Advances in neural information processing systems* (Vol. 36).

Haque, A., Tancik, M., Efros, A., Holynski, A., & Kanazawa, A. (2023). Instruct-nerf2nerf: Editing 3d scenes with instructions. In *Proceedings of the ieee/cvf international conference on computer vision.*

Hedman, P., Srinivasan, P.P., Mildenhall, B., Barron, J.T., & Debevec, P. (2021). Baking neural radiance fields for real-time view synthesis. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 5875–5884).

Hertz, A., Aberman, K., & Cohen-Or, D. (2023). Delta denoising score. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 2328–2337).

Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., & Cohen-Or, D. (2023). Prompt-to-prompt image editing with cross attention control. In *International conference on learning representations.*

Hertz, A., Perel, O., Giryes, R., Sorkine-Hornung, O., & Cohen-Or, D. (2022). Spaghetti: Editing implicit shapes through part aware generation. *ACM Transactions on Graphics (TOG), 41*(4), 1–20. https://doi.org/10.1145/3528223.3530084

Hertz, A., Voynov, A., Fruchter, S., & Cohen-Or, D. (2024). Style aligned image generation via shared attention. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 4775–4785).

Hertzmann, A. (1998). Painterly rendering with curved brush strokes of multiple sizes. In *Proceedings of the 25th annual conference on computer graphics and interactive techniques* (pp. 453–460).

Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., & Salesin, D.H. (2001). Image analogies. In *Proceedings of the 28th annual conference on computer graphics and interactive techniques* (pp. 327–340).

Hessel, J., Holtzman, A., Forbes, M., Bras, R.L., & Choi, Y. (2021). Clipscore: A reference-free evaluation metric for image captioning. arXiv:2104.08718

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems* (Vol. 30, pp. 6626–6637).

Hoffman, J., Hu, T., Kanyuk, P., Marshall, S., Nguyen, G., Schroers, H., & Witting, P. (2023). Creating elemental characters: From sparks to fire. *Acm siggraph 2023 talks* (pp. 1–2).

Höllein, L., Johnson, J., & Nießner, M. (2022). Stylemesh: Style transfer for indoor 3d scene reconstructions. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 6198–6208).

Houzz (2024). *Houzz.* https://www.houzz.com/

Hu, E., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., & Chen, W. (2022). LoRA: Low-rank adaptation of large language models. *International conference on learning representations.* https://openreview.net/forum?id=nZeVKeeFYf9

Hu, J., Hui, K.-H., Liu, Z., Li, R., & Fu, C.-W. (2024). Neural wavelet-domain diffusion for 3d shape generation, inversion, and manipulation. *ACM Trans. Graph.* https://doi.org/10.1145/3635304

Huang, H., Chen, Y., Zhang, T., & Yeung, S.-K. (2022). Real-time omnidirectional roaming in large scale indoor scenes. *SIGGRAPH Asia 2022 Technical Communications*, 1–5, https://doi.org/10.1145/3550340.3564222

Huang, H., Tseng, H.-Y., Saini, S., Singh, M., & Yang, M.-H. (2021). Learning to stylize novel views. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 13869–13878).

Huang, X., & Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the ieee international conference on computer vision* (pp. 1501–1510).

Huang, X., Liu, M.-Y., Belongie, S., & Kautz, J. (2018). Multimodal unsupervised image-to-image translation. In *Proceedings of the european conference on computer vision (eccv)* (pp. 172–189).

Huang, Y., He, Y., Yuan, Y.-J., Lai, Y.-K., & Gao, L. (2022). Stylized-NeRF: consistent 3d scene stylization as stylized nerf via 2d-3d mutual learning. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 18342–18352).

Hui, K., Li, R., Hu, J., & Fu, C.-W. (2022). Neural wavelet-domain diffusion for 3d shape generation. In *Siggraph asia 2022 conference papers.*

Hwang, I., Kim, H., & Kim, Y.M. (2023). Text2scene: Text-driven indoor scene stylization with part-aware details. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 1890–1899).

Ibrahimli, N., Kooij, J.F.P., & Nan, L. (2024). Muviecast: Multi-view consistent artistic style transfer. *3dv.*

Ioannou, E., & Maddock, S. (2023). Neural style transfer for computer games. *Bmvc workshop proceedings.*

Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A.A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1125–1134).

Jamriška, O., Sochorová, Š, Texler, O., Lukáč, M., Fišer, J., Lu, J., & Sỳkora, D. (2019). Stylizing video by example. *ACM Transactions on Graphics (TOG), 38*(4), 1–11. https://doi.org/10.1145/3306346.3323006

Jang, W., Ju, G., Jung, Y., Yang, J., Tong, X., & Lee, S. (2021). Stylecarigan: caricature generation via stylegan feature map modulation. *ACM Transactions on Graphics (TOG), 40*(4), 1–16. https://doi.org/10.1145/3450626.3459860

Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanæs, H. (2014). Large scale multi-view stereopsis evaluation. In *2014 ieee conference on computer vision and pattern recognition* (pp. 406–413).

Jiang, W., Trulls, E., Hosang, J., Tagliasacchi, A., Yi, K.M. (2021). Cotr: Correspondence transformer for matching across images. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 6207–6217).

Jing, Y., Yang, Y., Feng, Z., Ye, J., Yu, Y., & Song, M. (2019). Neural style transfer: A review. *IEEE Transactions on Visualization and Computer Graphics, 26*(11), 3365–3385. https://doi.org/10.1109/TVCG.2019.2921336

Johnson, J., Alahi, A., & Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision* (pp. 694–711).

Joshi, B., Stewart, K., & Shapiro, D. (2017). Bringing impressionism to life with neural style transfer in come swim. In *Proceedings of the acm siggraph digital production symposium* (pp. 1–5).

Kanyuk, P., Azevedo, V., Ortiz, R., & Tang, J. (2023). Singed silhouettes and feed forward flames: Volumetric neural style transfer for expressive fire simulation. *Acm siggraph 2023 talks* (pp. 1–2).

Kato, H., Ushiku, Y., & Harada, T. (2018). Neural 3d mesh renderer. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3907–3916).

Kerbl, B., Kopanas, G., Leimkühler, T., & Drettakis, G. (2023). 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (ToG), 42*(4), 1–14. https://doi.org/10.1145/3592433

Kim, B., Azevedo, V. C., Gross, M., & Solenthaler, B. (2019). Transport-based neural style transfer for smoke simulations. *ACM Transactions on Graphics (TOG).* https://doi.org/10.1145/3355089.3356560

Kim, B., Azevedo, V. C., Gross, M., & Solenthaler, B. (2020). Lagrangian neural style transfer for fluids. *ACM Transactions on Graphics (TOG), 39*(4), 52–1. https://doi.org/10.1145/3386569.3392473

Kim, G., Youwang, K., & Oh, T.-H. (2024). Fprf: Feed-forward photorealistic style transfer of large-scale 3d neural radiance fields. In *Proceedings of the aaai conference on artificial intelligence.*

Kim, J., Hua, B.-S., Nguyen, T., & Yeung, S.-K. (2023). Pointinverter: Point cloud reconstruction and editing via a generative model with shape priors. In *Proceedings of the ieee/cvf winter conference on applications of computer vision* (pp. 592–601).

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., & Girshick, R. (2023). Segment anything. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)* (p.4015–4026).

Knapitsch, A., Park, J., Zhou, Q.-Y., & Koltun, V. (2017). Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics. doi, 10*(1145/3072959), 3073599.

Knodt, J., & Gao, X. (2023). Consistent mesh diffusion. *arXiv preprint*, 1–17, https://doi.org/10.48550/arXiv.2312.00971

Kolkin, N., Kucera, M., Paris, S., Sykora, D., Shechtman, E., & Shakhnarovich, G. (2022). Neural neighbor style transfer.

Koo, J., Huang, I., Achlioptas, P., Guibas, L.J., & Sung, M. (2022). Partglot: Learning shape part segmentation from language reference games. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 16505–16514).

Koo, J., Park, C., & Sung, M. (2024). Posterior distillation sampling. *Cvpr.*

Koo, J., Yoo, S., Nguyen, M.H., & Sung, M. (2023). Salad: Part-level latent diffusion for 3d shape generation and manipulation. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 14441–14451).

Kotovenko, D., Grebenkova, O., Sarafianos, N., Paliwal, A., Ma, P., Poursaeed, O., & others (2024). Wast-3d: Wasserstein-2 distance for scene-to-scene stylization on 3d gaussians. *European conference on computer vision* (pp. 298–314).

Kwon, G., & Ye, J.C. (2022). Clipstyler: Image style transfer with a single text condition. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 18062–18071).

Lai, W., Huang, J.-B., Wang, O., Shechtman, E., Yumer, E., & Yang, M.-H. (2018). Learning blind video temporal consistency. In *Proceedings of the european conference on computer vision (eccv)* (pp. 170–185).

Laine, S., Hellsten, J., Karras, T., Seol, Y., Lehtinen, J., & Aila, T. (2020). Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics. doi, 10*(1145/3414685), 3417861.

Lee, H., Tseng, H.-Y., Huang, J.-B., Singh, M., & Yang, M.-H. (2018). Diverse image-to-image translation via disentangled representations. In *Proceedings of the european conference on computer vision (eccv)* (pp. 35–51).

Lei, J., Zhang, Y., & Jia, K., et al. (2022). Tango: Text-driven photorealistic and robust 3d stylization via lighting decomposition. *Advances in neural information processing systems* (Vol. 35, pp. 30923–30936).

Li, C., & Wand, M. (2016). Combining markov random fields and convolutional neural networks for image synthesis. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 2479–2486).

Li, J., Li, D., Savarese, S., & Hoi, S. (2023). Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning* (pp. 19730–19742).

Li, K., Fan, Y., Wu, Y., Sun, Z., Yang, W., Ji, X., & Chen, J. (2025). Learning pseudo 3d guidance for view-consistent texturing with 2d diffusion. In *European conference on computer vision* (pp. 18–34).

Li, R., Li, X., Hui, K.-H., & Fu, C.-W. (2021). Sp-gan: Sphere-guided 3d shape generation and manipulation. *ACM Transactions on Graphics (TOG), 40*(4), 1–12. https://doi.org/10.1145/3450626.3459766

Li, W., Chen, R., Chen, X., & Tan, P. (2024). Sweetdreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d. *The twelfth international conference on learning representations.* https://openreview.net/forum?id=extpNXo6hB

Li, X., Cao, Z., Wu, Y., Wang, K., Xian, K., Wang, Z., & Lin, G. (2024). S-dyrf: Reference-based stylized radiance fields for dynamic scenes. *Cvpr.*

Li, X., Liu, S., Kautz, J., & Yang, M.-H. (2019). Learning linear transformations for fast image and video style transfer. In *Proceedings*

of the ieee/cvf conference on computer vision and pattern recognition (pp. 3809–3817).

Li, Y., Jiang, L., Xu, L., Xiangli, Y., Wang, Z., Lin, D., & Dai, B. (2023). Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 3205–3215).

Li, Y., Lin, Z.-H., Forsyth, D., Huang, J.-B., & Wang, S. (2023). Climatenerf: Extreme weather synthesis in neural radiance field. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 3227–3238).

Liao, P., Li, X., Liu, X., & Keutzer, K. (2022). The artbench dataset: Benchmarking generative models with artworks. 1–26, https://doi.org/10.48550/arXiv.2206.11404

Lin, C., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., & Lin, T.-Y. (2023). Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 300–309).

Lin, T., Maire, M., Belongie, b., Hays, J., Perona, P., Ramanan, D., & Zitnick, C.L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740–755).

Lindholm, E., Kilgard, M.J., & Moreton, H. (2001). A user-programmable vertex engine. In *Proceedings of the 28th annual conference on computer graphics and interactive techniques* (pp. 149–158).

Liu, H., Tao, M., & Jacobson, A. (2018). Paparazzi: surface editing by way of multi-view image processing. *ACM Transactions on Graphics (TOG), 37*(6), 221–1. https://doi.org/10.1145/3272127.3275047

Liu, K., Zhan, F., Chen, Y., Zhang, J., Yu, Y., Saddik, A.E., & Xing, E. (2023). Stylerf: Zero-shot 3d style transfer of neural radiance fields. In *Proc. ieee conf. on computer vision and pattern recognition (cvpr).*

Liu, K., Zhan, F., Xu, M., Theobalt, C., Shao, L., & Lu, S. (2024). Stylegaussian: Instant 3d style transfer with gaussian splatting. arxiv:2403.07807

Liu, M., Breuel, T., & Kautz, J. (2017). Unsupervised image-to-image translation networks. *Advances in neural information processing systems* (pp. 700–708).

Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., & Vondrick, C. (2023). Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 9298–9309).

Liu, S., Lin, T., He, D., Li, F., Wang, M., Li, X., & Ding, E. (2021). Adaattn: Revisit attention mechanism in arbitrary neural style transfer. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 6649–6658).

Liu, Y., Fan, B., Meng, G., Lu, J., Xiang, S., & Pan, C. (2019). Densepoint: Learning densely contextual representation for efficient point cloud processing. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 5239–5248).

Liu, Y., Zhu, J., Tang, J., Zhang, S., Zhang, J., Cao, W., & Huang, D. (2024). Texdreamer: Towards zero-shot high-fidelity 3d human texture generation. In *European conference on computer vision.*

Liu, Z., Tang, H., Lin, Y., & Han, S. (2019). Point-voxel cnn for efficient 3d deep learning. *Advances in neural information processing systems* (Vol. 32).

Logacheva, E., Suvorov, R., Khomenko, O., Mashikhin, A., & Lempitsky, V. (2020). Deeplandscape: Adversarial modeling of landscape videos. In *Computer vision–eccv 2020: 16th european conference, glasgow, uk, august 23–28, 2020, proceedings, part xxiii 16* (pp. 256–272).

Long, X., Guo, Y.-C., Lin, C., Liu, Y., Dou, Z., Liu, L., & others (2024). Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Lorensen, W., & Cline, H. E. (1987). Marching cubes: A high resolution 3d surface construction algorithm. *Acm Siggraph Computer Graphics. doi, 10*(1145/37401), 37422.

Luan, F., Paris, S., Shechtman, E., & Bala, K. (2017). Deep photo style transfer. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4990–4998).

Lüddecke, T., & Ecker, A. (2022). Image segmentation using text and image prompts. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 7086–7096).

Luma (2023). *Luma ai.* https://lumalabs.ai/

Ma, Y., Zhang, X., Sun, X., Ji, J., Wang, H., Jiang, G., & Ji, R. (2023). X-mesh: Towards fast and accurate text-driven 3d stylization via dynamic textual guidance. In *roceedings of the ieee/cvf international conference on computer vision (iccv).*

Manzaneque, F.R. (2023). *Revolutionizing virtual production: How neural radiance fields will supercharge production pipelines: Neural radiance fields.* https://neuralradiancefields.io/revolutionizing-virtual-production-how-neural-radiance-fields-will-supercharge-production-pipelines/

Martin-Brualla, R., Radwan, N., Sajjadi, M.S., Barron, J.T., Dosovitskiy, A., & Duckworth, D. (2021). Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 7210–7219).

Masci, J., Boscaini, D., Bronstein, M., & Vandergheynst, P. (2015). Geodesic convolutional neural networks on riemannian manifolds. In *Proceedings of the ieee international conference on computer vision workshops* (pp. 37–45).

McGuire, M., Ekanayake, C., St-Amour, J.-F., Halén, H., Thibault, A., & Martel, B. (2010). Stylized rendering in games. *Acm siggraph courses.* https://stylized.realtimerendering.com/

Mei, Y., Xu, J., & Patel, V.M. (2024). ReGS: Reference-based controllable scene stylization with gaussian splatting. *The thirty-eighth annual conference on neural information processing systems.* https://openreview.net/forum?id=ynJr0RW6FR

Menapace, W., Lathuilière, S., Siarohin, A., Theobalt, C., Tulyakov, S., Golyanik, V., & Ricci, E. (2022). Playable environments: Video manipulation in space and time. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 3584–3593).

Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., & Geiger, A. (2019). Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 4460–4470).

Meshy (2024). *Meshy: Create stunning 3d models with ai.* https://www.meshy.ai/

Michel, O., Bar-On, R., Liu, R., Benaim, S., & Hanocka, R. (2022). Text2mesh: Text-driven neural stylization for meshes. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Midjourney (2023). *Midjourney.* https://www.midjourney.com/

Mildenhall, B., Srinivasan, P. P., Ortiz-Cayon, R., Kalantari, N. K., Ramamoorthi, R., Ng, R., & Kar, A. (2019). Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG), 38*(4), 1–14. https://doi.org/10.1145/3306346.3322980

Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., & Ng, R. (2020). Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision* (pp. 405–421).

Müller, T., Evans, A., Schied, C., & Keller, A. (2022). Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG), 41*(4), 1–15. https://doi.org/10.1145/3528223.3530127

Navarro, M., & Rice, J. (2021). Stylizing volumes with neural networks. *Acm siggraph 2021 talks* (pp. 1–2).

Nguyen-Phuoc, T., Liu, F., & Xiao, L. (2022). Snerf: stylized neural implicit representations for 3d scenes. In *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2022* , https://doi.org/10.1145/3528223.3530107

Nguyen-Phuoc, T., Richardt, C., Mai, L., Yang, Y., & Mitra, N. (2020). Blockgan: Learning 3d object-aware scene representations from unlabelled images. *Advances in Neural Information Processing Systems, 33*, 6767–6778.

Nichol, A., Jun, H., Dhariwal, P., Mishkin, P., & Chen, M. (2022). Point-e: A system for generating 3d point clouds from complex prompts. 1–14, https://doi.org/10.48550/arXiv.2212.08751

Niemeyer, M., & Geiger, A. (2021). Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 11453–11464).

Oechsle, M., Mescheder, L., Niemeyer, M., Strauss, T., & Geiger, A. (2019). Texture fields: Learning texture representations in function space. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 4531–4540).

OpenAI (2023). *Dall · e*3. https://openai.com/dall-e-3

OpenAI (2024). *Sora: Creating video from text.* https://openai.com/sora

Orghidan, R., Bedenk, T., & Wegner, K. (2022). *An r& d project on ai for in 3d asset creation for games.* Endava. https://www.endava.com/en/blog/Engineering/2022/An-R-D-Project-on-AI-in-3D-Asset-Creation-for-Games

Painter by numbers. (2016). *Kaggle*, https://www.kaggle.com/c/painter-by-numbers

Pang, H., Hua, B.-S., & Yeung, S.-K. (2023). Locally stylized neural radiance fields. In *Proceedings of the ieee/cvf international conference on computer vision (iccv).*

Park, J., Florence, P., Straub, J., Newcombe, R., & Lovegrove, S. (2019). Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 165–174).

Park, J., Kwon, G., & Ye, J.C. (2024). Ed-nerf: Efficient text-guided editing of 3d scene using latent space nerf. In *International conference on learning representations.*

Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., & Martin-Brualla, R. (2021). Nerfies: Deformable neural radiance fields. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 5865–5874).

Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., & Lischinski, D. (2021). Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 2085–2094).

Peng, S., Niemeyer, M., Mescheder, L., Pollefeys, M., & Geiger, A. (2020). Convolutional occupancy networks. In *Computer vision–eccv 2020: 16th european conference, glasgow, uk, august 23–28, 2020, proceedings, part iii 16* (pp. 523–540).

Pérez, J., Nguyen-Phuoc, T., Cao, C., Sanakoyeu, A., Simon, T., Arbeláez, P., & Pumarola, A. (2024). Styleavatar: Stylizing animatable head avatars. In *Proceedings of the ieee/cvf winter conference on applications of computer vision* (pp. 8678–8687).

Perla, S.R.K., Wang, Y., Mahdavi-Amiri, A., & Zhang, H. (2024). Easi-tex: Edge-aware mesh texturing from single image. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, *43*(4), , https://doi.org/10.1145/3658222

Pfister, H., Zwicker, M., Van Baar, J., & Gross, M. (2000). Surfels: Surface elements as rendering primitives. In *Proceedings of the 27th annual conference on computer graphics and interactive techniques* (pp. 335–342).

Planner5D (2024). *Planner5d.* https://planner5d.com/

Po, R., Yifan, W., Golyanik, V., Aberman, K., Barron, J.T., Bermano, A.H., & Wetzstein, G. (2023). State of the art on diffusion models for visual computing. 1–32, https://doi.org/10.48550/arXiv.2310.07204

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., & Rombach, R. (2023). Sdxl: Improving latent diffusion models for high-resolution image synthesis. arXiv:2307.01952

Poole, B., Jain, A., Barron, J.T., & Mildenhall, B. (2023). Dreamfusion: Text-to-3d using 2d diffusion. In *The eleventh international conference on learning representations.*

Qi, C., Su, H., Mo, K., & Guibas, L.J. (2017). Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 652–660).

Qi, T., Fang, S., Wu, Y., Xie, H., Liu, J., Chen, L., & Zhang, Y. (2024). Deadiff: An efficient stylization diffusion model with disentangled representations. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 8693–8702).

Qian, G., Mai, J., Hamdi, A., Ren, J., Siarohin, A., Li, B., & Ghanem, B. (2024). Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. In *The twelfth international conference on learning representations.* https://openreview.net/forum?id=0jHkUDyEO9

Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., & others (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748–8763).

Ravi, N., Reizenstein, J., Novotny, D., Gordon, T., Lo, W.-Y., Johnson, J., & Gkioxari, G. (2020). Accelerating 3d deep learning with pytorch3d. 1–18, https://doi.org/10.48550/arXiv.2007.08501

Renderpeople (2023). *Renderpeople.* https://renderpeople.com/

Richardson, E., Metzer, G., Alaluf, Y., Giryes, R., & Cohen-Or, D. (2023). Texture: Text-guided texturing of 3d shapes. In *Acm siggraph 2023 conference proceedings.* New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3588432.3591503

Riegler, G., & Koltun, V. (2020). Free view synthesis. In *European conference on computer vision* (pp. 623–640).

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 10684–10695).

Ruder, M., Dosovitskiy, A., & Brox, T. (2016). Artistic style transfer for videos. In *Pattern recognition: 38th german conference, gcpr 2016, hannover, germany, september 12-15, 2016, proceedings 38* (pp. 26–36).

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K. (2023). Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 22500–22510).

Runway (2023). *Runway.* https://runwayml.com/

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., & others (2022). Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in neural information processing systems* (Vol. 35, pp. 36479–36494).

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. (2016). Improved techniques for training gans. In *Advances in neural information processing systems* (Vol. 29, pp. 2234–2242).

Schwarz, K., Liao, Y., Niemeyer, M., & Geiger, A. (2020). Graf: Generative radiance fields for 3d-aware image synthesis. *Advances in neural information processing systems* (Vol. 33, pp. 20154–20166).

Sella, E., Fiebelman, G., Atia, N., & Averbuch-Elor, H. (2023). Spic-e: Structural priors in 3d diffusion models using cross-entity attention. arXiv:2311.17834

Sella, E., Fiebelman, G., Hedman, P., & Averbuch-Elor, H. (2023). Vox-e: Text-guided voxel editing of 3d objects. In *Proceedings of the*

ieee/cvf international conference on computer vision (pp. 430–440).

Shih, Y., Paris, S., Durand, F., & Freeman, W. T. (2013). Data-driven hallucination of different times of day from a single outdoor photo. *ACM Transactions on Graphics (TOG), 32*(6), 1–11.

Shum, K.C., Kim, J., Hua, B.-S., Nguyen, D.T., & Yeung, S.-K. (2024). Language-driven object fusion into neural radiance fields with pose-conditioned dataset updates. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Sidi, O., Van Kaick, O., Kleiman, Y., Zhang, H., Cohen-Or, D. (2011). Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. In *Proceedings of the 2011 siggraph asia conference* (pp. 1–10).

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International conference on learning representations.*

Singh, A., Jaiswal, V., Joshi, G., Sanjeeve, A., Gite, S., & Kotecha, K. (2021). Neural style transfer: A critical review. *IEEE Access*.

Song, G., Luo, L., Liu, J., Ma, W.-C., Lai, C., Zheng, C., & Cham, T.-J. (2021). Agilegan: stylizing portraits by inversion-consistent transfer learning. *ACM Transactions on Graphics (TOG), 40*(4), 1–13. https://doi.org/10.1145/3450626.3459771

Song, H., Choi, S., Do, H., Lee, C., & Kim, T. (2023). Blending-nerf: Text-driven localized editing in neural radiance fields. In *Proceedings of the ieee/cvf international conference on computer vision.*

Song, L., Chen, A., Li, Z., Chen, Z., Chen, L., Yuan, J., & Geiger, A. (2023). Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields. *IEEE Transactions on Visualization and Computer Graphics, 29*(5), 2732–2742.

Sorkine, O., & Alexa, M. (2007). As-rigid-as-possible surface modeling. In *Symposium on geometry processing* (Vol. 4, pp. 109–116).

Spline (2023). *Splineai.* https://spline.design/ai-style-transfer

Srinivasan, P., Deng, B., Zhang, X., Tancik, M., Mildenhall, B., & Barron, J.T. (2021). Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 7495–7504).

Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., & others (2019). The replica dataset: A digital replica of indoor spaces. arXiv:1906.05797, 1–10, https://doi.org/10.48550/arXiv.1906.05797

Sumner, R., Schmid, J., & Pauly, M. (2007). Embedded deformation for shape manipulation. *Acm siggraph 2007 papers* (pp. 80–es).

Sun, C., Sun, M., & Chen, H.-T. (2022). Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 5459–5469).

Sun, J., Zhang, B., Shao, R., Wang, L., Liu, W., Xie, Z., & Liu, Y. (2024). Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior. In *International conference on learning representations.*

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 2818–2826).

Tancik, M., Casser, V., Yan, X., Pradhan, S., Mildenhall, B., Srinivasan, P.P., & Kretzschmar, H. (2022). Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 8248–8258).

Tancik, M., Weber, E., Ng, E., Li, R., Yi, B., Wang, T., & others (2023). Nerfstudio: A modular framework for neural radiance field development. In *Acm siggraph 2023 conference proceedings* (pp. 1–12).

Taniguchi, D. (2019). Neural ar: immersive augmented reality with real-time neural style transfer. *Acm siggraph 2019 virtual, augmented, and mixed reality* (pp. 1–1).

Teed, Z., & Deng, J. (2020). Raft: Recurrent all-pairs field transforms for optical flow. In *Computer vision–eccv 2020: 16th european conference, glasgow, uk, august 23–28, 2020, proceedings, part ii 16* (pp. 402–419).

Tewari, A., Fried, O., Thies, J., Sitzmann, V., Lombardi, S., Sunkavalli, K., & others (2020). State of the art on neural rendering. *Computer graphics forum* (Vol. 39, pp. 701–727).

Tewari, A., Thies, J., Mildenhall, B., Srinivasan, P., Tretschk, E., Yifan, W., & others (2022). Advances in neural rendering. *Computer graphics forum* (pp. 703–735).

Thies, J., Zollhöfer, M., & Nießner, M. (2019). Deferred neural rendering: Image synthesis using neural textures. *Acm Transactions on Graphics (TOG), 38*(4), 1–12. https://doi.org/10.1145/3306346.3323035

Tseng, K., Huang, J.-Y., Chen, Y.-S., Chen, C.-S., & Hung, Y.-P. (2022). Pseudo-3d scene modeling for virtual reality using stylized novel view synthesis. *Acm siggraph 2022 posters.*

Tumanyan, N., Bar-Tal, O., Bagon, S., & Dekel, T. (2022). Splicing vit features for semantic appearance transfer. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 10748–10757).

TurboSquid (2023). *Turbosquid.* https://www.turbosquid.com/

Turki, H., Zhang, J.Y., Ferroni, F., & Ramanan, D. (2023). Suds: Scalable urban dynamic scenes. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 12375–12385).

Vachha, C., & Haque, A. (2024). *Instruct-gs2gs: Editing 3d gaussian splats with instructions.* https://instruct-gs2gs.github.io/

Volinga (2023). *Volinga.* https://volinga.ai/

Wang, C., Chai, M., He, M., Chen, D., & Liao, J. (2022). Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 3835–3844).

Wang, C., Jiang, R., Chai, M., He, M., Chen, D., & Liao, J. (2023). Nerf-art: Text-driven neural radiance fields stylization. *IEEE Transactions on Visualization and Computer Graphics*.

Wang, H., Wang, Q., Bai, X., Qin, Z., & Chen, A. (2024). Instantstyle: Free lunch towards style-preserving in text-to-image generation. *arXiv preprint* arXiv:2404.02733

Wang, Q., Wang, Z., Genova, K., Srinivasan, P.P., Zhou, H., Barron, J.T., & Funkhouser, T. (2021). Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 4690–4699).

Wang, Y., Cheng, J.-S., Feng, Q., Tao, W.-Y., Lai, Y.-K., & Li, K. (2023). Tsnerf: Text-driven stylized neural radiance fields via semantic contrastive learning. *Computers & Graphics, 116*, 102–114. https://doi.org/10.1016/j.cag.2023.08.009

Wang, Y., Yi, X., Wu, Z., Zhao, N., Chen, L., & Zhang, H. (2025). View-consistent 3d editing with gaussian splatting. In *European conference on computer vision* (pp. 404–420).

Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., & Zhu, J. (2023). Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in neural information processing systems.*

Wen, L., Gao, C., & Zou, C. (2023). Cap-vstnet: Content affinity preserved versatile style transfer. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 18300–18309).

Wu, J., Bian, J.-W., Li, X., Wang, G., Reid, I., Torr, P., & Prisacariu, V. (2024). GaussCtrl: Multi-view consistent text-driven 3D Gaussian splatting editing. *ECCV*.

Wu, T., Li, Z., Yang, S., Zhang, P., Pan, X., Wang, J., & Liu, Z. (2023). Hyperdreamer: Hyper-realistic 3d content generation and editing from a single image. In *Siggraph asia 2023 conference papers* (pp. 1–10).

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., & Xiao, J. (2015). 3d shapenets: A deep representation for volumetric shapes.

In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1912–1920).

Wu, Z., Zhu, Z., Du, J., & Bai, X. (2022). Ccpl: contrastive coherence preserving loss for versatile style transfer. In *European conference on computer vision* (pp. 189–206).

Xiangli, Y., Xu, L., Pan, X., Zhao, N., Rao, A., Theobalt, C., & Lin, D. (2022). Bungeenerf: Progressive neural radiance field for extreme multi-scale scene rendering. In *European conference on computer vision* (pp. 106–122).

Xie, T., Zong, Z., Qiu, Y., Li, X., Feng, Y., Yang, Y., & Jiang, C. (2024). Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Xie, Y., Takikawa, T., Saito, S., Litany, O., Yan, S., Khan, N., & Sridhar, S. (2022). Neural fields in visual computing and beyond. *Computer graphics forum* (pp. 641–676).

Xu, L., Xiangli, Y., Peng, S., Pan, X., Zhao, N., Theobalt, C., & Lin, D. (2023). Grid-guided neural radiance fields for large urban scenes. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 8296–8306).

Yang, B., Dong, W., Ma, L., Hu, W., Liu, X., Cui, Z., & Ma, Y. (2024). Dreamspace: Dreaming your room space with text-driven panoramic texture propagation. *Ieee vr.*

Yang, H., Chen, Y., Pan, Y., Yao, T., Chen, Z., & Mei, T. (2023). 3dstyle-diffusion: Pursuing fine-grained text-driven 3d stylization with 2d diffusion models. In *Proceedings of the 31st acm international conference on multimedia* (pp. 6860–6868).

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., & Yang, M.-H. (2023). Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*. https://doi.org/10.1145/3626235

Yang, S., Jiang, L., Liu, Z., & Loy, C.C. (2022). Pastiche master: Exemplar-based high-resolution portrait style transfer. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 7693–7702).

Yang, Z., Yang, H., Pan, Z., Zhu, X., & Zhang, L. (2024). Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International conference on learning representations (iclr).*

Yariv, L., Gu, J., Kasten, Y., & Lipman, Y. (2021). Volume rendering of neural implicit surfaces. *Advances in neural information processing systems* (Vol. 34, pp. 4805–4815).

Ye, H., Zhang, J., Liu, S., Han, X., & Yang, W. (2023). Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. arXiv:2308.06721

Ye, M., Danelljan, M., Yu, F., & Ke, L. (2025). Gaussian grouping: Segment and edit anything in 3d scenes. In *European conference on computer vision* (pp. 162–179).

Yeh, Y., Huang, J.-B., Kim, C., Xiao, L., Nguyen-Phuoc, T., Khan, N., & others (2024). Texturedreamer: Image-guided texture synthesis through geometry-aware diffusion. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Yi, L., Kim, V. G., Ceylan, D., Shen, I.-C., Yan, M., Su, H., & Guibas, L. (2016). A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (ToG), 35*(6), 1–12. https://doi.org/10.1145/2980179.2980238

Yin, K., Gao, J., Shugrina, M., Khamis, S., & Fidler, S. (2021). 3dstylenet: Creating 3d shapes with geometric and texture style variations. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 12456–12465).

Yoo, J., Uh, Y., Chun, S., Kang, B., & Ha, J.-W. (2019). Photorealistic style transfer via wavelet transforms. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 9036–9045).

Youwang, K., Oh, T.-H., & Pons-Moll, G. (2024). Paint-it: Text-to-texture synthesis via deep convolutional texture map optimization and physically-based rendering. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Yu, L., Yeung, S.K., Tang, C.K., Terzopoulos, D., Chan, T.F., & Osher, S.J. (2011). Make it home: automatic optimization of furniture arrangement. *ACM Transactions on Graphics (TOG)-Proceedings of ACM SIGGRAPH 2011, v. 30,(4), July 2011, article no. 86*, *30*(4), ,

Yu, X., Dai, P., Li, W., Ma, L., Liu, Z., & Qi, X. (2023). Texture generation on 3d meshes with point-uv diffusion. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 4206–4216).

Zeng, X., Chen, X., Qi, Z., Liu, W., Zhao, Z., Wang, Z., & Yu, G. (2024). Paint3d: Paint anything 3d with lighting-less texture diffusion models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition.*

Zhan, F., Yu, Y., Wu, R., Zhang, J., Lu, S., Liu, L., & Xing, E. (2023). *Multimodal image synthesis and editing: The generative ai era*. Ieee transactions on pattern analysis and machine intelligence: IEEE.

Zhang, D., Chen, Z., Yuan, Y.-J., Zhang, F.-L., He, Z., Shan, S., & Gao, L. (2024). Stylizedgs: Controllable stylization for 3d gaussian splatting. arXiv:2404.05220

Zhang, D., Fernandez-Labrador, C., & Schroers, C. (2024). Coarf: Controllable 3d artistic style transfer for radiance fields. In *2024 international conference on 3d vision (3dv)* (pp. 612–622).

Zhang, H., Feng, Y., Kulits, P., Wen, Y., Thies, J., & Black, M.J. (2024). Text-guided generation and editing of compositional 3d avatars. *3dv.*

Zhang, H., Pan, Z., Zhang, C., Zhu, L., & Gao, X. (2024). Texpainter: Generative mesh texturing with multi-view consistency. In *Acm siggraph 2024 conference papers* (pp. 1–11).

Zhang, J., Lan, Y., Yang, S., Hong, F., Wang, Q., Yeo, C.K., & Loy, C.C. (2023). Deformtoon3d: Deformable 3d toonification from neural radiance fields. *Iccv.*

Zhang, K., Kolkin, N., Bi, S., Luan, F., Xu, Z., Shechtman, E., & Snavely, N. (2022). Arf: Artistic radiance fields. *European conference on computer vision* (pp. 717–733).

Zhang, K., Riegler, G., Snavely, N., & Koltun, V. (2020). Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint*, 1–9, https://doi.org/10.48550/arXiv.2010.07492

Zhang, L., Rao, A., & Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 3836–3847).

Zhang, X., Yin, B.-W., Chen, Y., Lin, Z., Li, Y., Hou, Q., & Cheng, M.-M. (2024). Temo: Towards text-driven 3d stylization for multi-object meshes. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 19531–19540).

Zhang, Y., He, Z., Xing, J., Yao, X., & Jia, J. (2023). Ref-npr: Reference-based non-photorealistic radiance fields for controllable scene stylization. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 4242–4251).

Zhang, Y., Huang, N., Tang, F., Huang, H., Ma, C., Dong, W., & Xu, C. (2023). Inversion-based style transfer with diffusion models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 10146–10156).

Zhang, Y., Liu, Y., Xie, Z., Yang, L., Liu, Z., Yang, M., & Jin, X. (2024). Dreammat: High-quality pbr material generation with geometry- and light-aware diffusion models. In *ACM Trans. Graph.*, *43*(4), , https://doi.org/10.1145/3658170

Zhang, Z., Liu, Y., Han, C., Pan, Y., Guo, T., & Yao, T. (2023). Transforming radiance field with lipschitz network for photorealistic 3d scene stylization. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 20712–20721).

Zhao, H., Jiang, L., Jia, J., Torr, P.H., & Koltun, V. (2021). Point transformer. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 16259–16268).

Zhou, L., Du, Y., & Wu, J. (2021). 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 5826–5835).

Zhou, Q., & Jacobson, A. (2016). Thingi10k: A dataset of 10,000 3d-printing models. *arXiv preprint*, 1–8, https://doi.org/10.48550/arXiv.1605.04797

Zhu, J., Park, T., Isola, P., & Efros, A.A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the ieee international conference on computer vision* (pp. 2223–2232).

Zhuang, J., Kang, D., Cao, Y.-P., Li, G., Lin, L., & Shan, Y. (2024). Tip-editor: An accurate 3d editor following both text-prompts and image-prompts. *ACM Transactions on Graphics*, 1–11

Zhuang, J., Wang, C., Liu, L., Lin, L., & Li, G. (2023). Dreameditor: Text-driven 3d scene editing with neural fields. In *Acm siggraph asia 2023 conference proceedings.*