Make-It-Vivid: Dressing Your Animatable Biped Cartoon Characters from Text

Junshu Tang[1†] Yanhong Zeng[2] Ke Fan[1] Xuheng Wang[3]
Bo Dai[2] Kai Chen[2‡] Lizhuang Ma[1‡]
[1] Shanghai Jiao Tong University [2] Shanghai AI Lab [3] Tsinghua University

Abstract

*Creating and animating 3D biped cartoon characters is crucial and valuable in various applications. Compared with geometry, the diverse texture design plays an important role in making 3D biped cartoon characters vivid and charming. Therefore, we focus on automatic texture design for cartoon characters based on input instructions. This is challenging for domain-specific requirements and a lack of high-quality data. To address this challenge, we propose **Make-It-Vivid**, the first attempt to enable high-quality texture generation from text in UV space. We prepare a detailed text-texture paired data for 3D characters by using vision-question-answering agents. Then we customize a pretrained text-to-image model to generate texture map with template structure while preserving the natural 2D image knowledge. Furthermore, to enhance fine-grained details, we propose a novel adversarial learning scheme to shorten the domain gap between original dataset and realistic texture domain. Extensive experiments show that our approach outperforms current texture generation methods, resulting in efficient character texturing and faithful generation with prompts. Besides, we showcase various applications such as out of domain generation and texture stylization. We also provide an efficient generation system for automatic text-guided textured character generation and animation.*

1. Introduction

3D biped cartoon characters [28] breathe life into fictional characters, conveying actions, and storytelling elements engagingly. These characters find applications in various domains, including video games (*e.g.*, Animal Crossing [1]), movies (*e.g.*, Zootopia [58]), and the upcoming metaverse. Yet, the creation and animation of these characters heavily rely on skilled artists utilizing specialized software, making it a labor-intensive and time-consuming process.

Compared to the shape of 3D biped cartoon characters, their textures exhibit a significantly higher level of diversity, playing a crucial role in creating vivid and charming characters. This work focuses on the automatic design of high-quality textured characters by generating textures based on text descriptions, which presents two main challenges. **(1) Demanding domain-specific requirements.** Simply dressing 3D biped cartoon characters with appropriate textures is insufficient to make them attractive. These characters require textures that possess unique traits, including semantic harmony, consistent global configuration, and rich local high-frequency details. Consequently, conventional shape texturing methods are inadequate for cartoon characters, often resulting in textures with smooth and blurry details, as well as noticeable seam artifacts [6–8, 21, 29, 30, 41]. **(2) Limited availability of high-quality data.** The scarcity of high-quality data that meets the demanding requirements further complicates the task. The creation of high-quality cartoon characters involves a costly and skill-intensive pro-

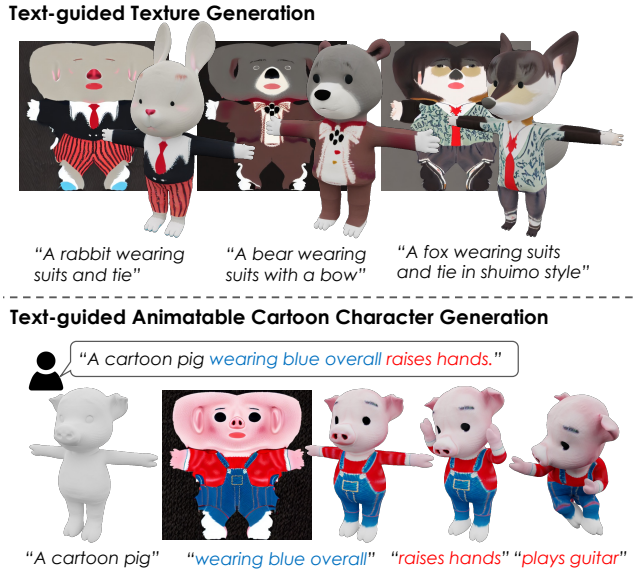

Figure 1. We present *Make-it-Vivid*, the first attempt that can create plausible and consistent texture in UV space for 3D biped cartoon characters from text input within few seconds. *Make-it-vivid* enables texture generation with fine-grained details in multiple styles (see above), and also supports efficient text-guided animatable textured character production (see bottom).

[†]Work done when interning at Shanghai AI Lab.
[‡]Corresponding authors.

cess, resulting in limited availability of such data. Additionally, due to intellectual property (IP) concerns, these data are often kept private, making it impractical to gather them from publicly accessible sources on the Internet. Existing datasets [28] that include character texture data also suffer from significant limitations in terms of high-frequency details, inter-instance variations, and paired text descriptions.

In this work, we introduce **Make-It-Vivid**, a novel texture generation framework specifically designed for 3D biped cartoon characters. Our framework enables the generation of diverse, high-fidelity, and visually compelling textures in a single forward pass, given text input. To address the challenge of limited high-quality data, we propose marrying a knowledgeable pre-trained text-to-image (T2I) diffusion model with a topology-aware representation of the UV space, making Make-It-Vivid the first framework to leverage diffusion priors in the UV space for 3D biped cartoon characters. We start by developing a specialized multi-agent-based captioning system tailored for 3D biped characters. By utilizing vision-question-answering agents, we can easily generate high-quality descriptions of color, clothing, and character types based on rendered frontal views for the UV maps. This process results in a dataset of high-quality text-UVMap pairs. Once the dataset is prepared, we customize the pre-trained T2I model to generate high-quality UV maps. This customization involves introducing learnable parameters and fine-tuning them on the paired text-UVMap data while keeping the T2I model fixed to retain its open-domain knowledge. This design allows our framework to seamlessly integrate with various customized T2I style models, such as Shuimo style [33] and American comics, for creative texturing.

While the customized diffusion model generates various plausible textures faithful to text prompts, the texture quality often suffers from over-smoothing, making it challenging to meet demanding domain-specific requirements. This limitation can be attributed to the lack of high-frequency details in the training data. Therefore, we create high-quality images using a T2I model as a proxy for high-frequency details and innovatively introduce adversarial training [13] into the diffusion training process, leading to enhanced texture details. We extensively evaluate the performance of Make-It-Vivid, demonstrating its superiority in texturing 3D biped characters. Our main contributions are as follows:

- We present **Make-It-Vivid**, which empowers non-expert users to effortlessly customize vivid 3D textured characters with desired identities, styles, and attributes.
- To overcome the limitation of training data, we are the first to introduce adversarial training into the diffusion training process, achieving improved image fidelity.
- We showcase the versatility of our approach by exploring captivating applications in stylized generation and multi-modality textured character animation.

2. Related Work

3D Generation under Text Guidance. Recent advancements in image generation [9, 10, 17, 37, 39, 40, 42, 43, 53] have greatly boosted the research progress in 3D assets generation [8, 16, 25, 38, 49, 50] under text guidance. A set of works [16, 30, 38, 45, 50] propose to generate 3D shapes by optimizing a NeRF representation [2, 34], either through CLIP guidance [16] or Score Distillation Sampling [38] and Variational Score Distillation [50] via 2D diffusion models. Though effective, implicit representations such as NeRF can be infeasible to be deployed for most practical applications [22, 47]. Subsequent methods [25, 32, 47] tackle the above problem by directly generating highly realistic 3D meshes from textual prompts. Specifically, Magic3D [25] presents a two-stage optimization framework to address the efficiency and resolution problems observed in NeRF-based models, while TextMesh [47] employs an SDF backbone to extract realistic-looking 3D meshes. There are also a number of works trying to address the different aspects of 3D content generation. For example, Fantasia3D [8] utilizes a hybrid representation of 3D shape, namely DMTet [44], and decouples the problem to geometry and appearance modeling. Point·E [35] presents an alternative approach to 3D content generation by utilizing a point cloud diffusion model. Latent-NeRF [30] optimizes SDS loss in Stable Diffusion's latent space to allow increased control over the generation process. Besides, [30] also presents Latent-Paint, which optimizes neural texture maps based on the input mesh.

3D Mesh Texturing under Text Guidance. In addition to generate fully textured shapes, texture generation based on the given 3D geometry has recently gained significant popularity. A number of works [8, 21, 29–31, 38] approach this task by optimizing an implicit representation of both 3D geometry and texture. For example, Text2Mesh [31], Tango [21] and XMesh [29] innovate 3D mesh texturing by optimizing the color and displacement for each vertex on the base mesh based on corresponding text prompt using CLIP loss [39]. While other methods [8, 25, 30, 38, 48, 50] leverage SDS loss which helps create high-fidelity and realistic texture. Another set of works introduce an iterative painting scheme to paint a given 3D model from different view points [6, 7, 41]. These methods synthesize multi-view textures based on observations under different viewpoints, and use depth-aware texture generation and inpainting to refine the new unpainted areas while preserving consistent texture from the partially painted area. Albeit improving results, these works still suffer from severe inconsistencies across multiple views and seam artifacts due to their inpainting nature. Furthermore, some of generative models focus on generating high-fidelity UV textures [11, 12, 18–20, 52, 54] directly, which shows impressive quality in 3D face reconstruction and generation. In this paper, we ex-

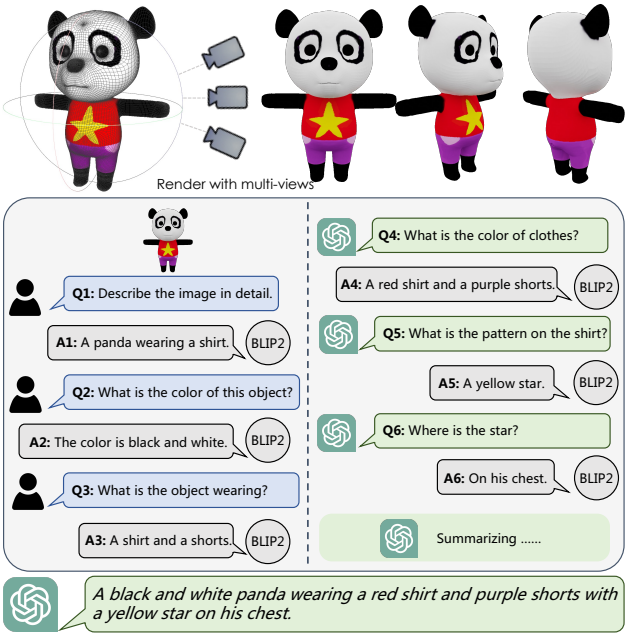

Figure 2. Multi-rounds of dialogue for captioning 3D characters. For each rendered image, we hard code three questions and use ChatGPT for asking follow-up three questions, then summarize.

plore UV texture generation on a more challenging but crucial scenario on vivid texture generation.

3. Preliminary

Parametric models of shapes, like 3DMM [4], FLAME [24] and SMPL [26], are widely used in computer graphics, computer vision, and other related fields. These models are designed to represent the 3D shape and appearance of complex objects, such as human bodies and faces, in a compact and expressive manner.

In order to represent 3D cartoon biped characters with a consistent topology, we adopt a parametric model Rabit [28] which contains a linear blend model for shapes. Specifically, the 3D biped character is parameterized as $M = F(B, \Theta, Z)$, where B denotes the identity-related body parameter, Θ represents the non-rigid pose-related parameter, and Z represents the latent embedding of texture.

In detail, the generated character shape is defined as $M_S = \bar{M}_S + \sum_i^{|B|} \beta_i s_i$, where the mean shape is denoted as $\bar{M}_S$ and $|B|$ denotes the dimension of shape coefficients. $s_i \in \mathbb{R}^{3*N}$ denotes the orthogonal principal components of vertex displacements of the geometry shape. The coefficients models the variants of shapes under different identities. Besides, the shape of eyeballs can be calculated based on the predefined landmarks.

The pose parameters $\Theta = [\theta_1, \theta_2, ..., \theta_K] \in \mathbb{R}^{69}$ denotes the axis-angle of the relative rotation of joint k with respect to its parents. $K = 23$ denotes the number of the joints. Each parameter θ_k can be converted to the rotation matrix using Rodrigues' formular:

$$
\begin{aligned}
v_i' &= \sum_{k=1}^{K} w_{k,i} G_k'(\theta, J) v_i, \\
G_k'(\Theta, J) &= G_k(\Theta, J) G_k(\Theta', J)^{-1}, \\
G_k(\Theta, J) &= \prod_{j \in A(k)} \begin{bmatrix} R(\theta_j) & J_j \\ 0 & 1 \end{bmatrix},
\end{aligned} \tag{1}
$$

where $w_{k,j}$ denotes the skinning weight for the i-th vertex. $G_k(\Theta, J)$ is the global transformation of joint k. J_j denotes the location of the j-th joint. We use this representation to animate the mesh using specific pose parameters Θ.

As for parametric texture embedding, Rabit uses a StyleGAN2-based [17] generator for embedding texture map to latent codes. Specifically, the texture image $T \in \mathbb{R}^{H \times W \times 3}$ is generated by a latent code $Z \in \mathbb{R}^d$ where the resolution is 1024 and the dimension $d = 512$. However, it can only generate textures unconditionally with low quality. In this paper, we propose a new text-driven vivid texture generator which is editable with multiple concepts, such as color, clothes, style and so on.

4. Text-guided UV Texture Generation

Texturing on the non-rigid cartoon character under simple instructions is crucial yet inherently challenging. We therefore propose the first attempt to prioritize the generation of texture maps using UV unwrapping, a consistent and essential representation for mesh textures in the traditional computer graphics pipeline. Drawing inspiration from the achievements in image synthesis through text-conditional diffusion models, we have adopted a diffusion model for generating textures from random noise conditioned by text.

To this end, given a user prompt P, our method is able to generate a vivid and consistent texture map aware of the definition of the correspondence between UV space and the geometry. Inspired by the appealing results from the pretrained latent diffusion models, which is trained on large and diverse text-image pairs, we leverage these knowledge as semantic priors for our texture generation. To enforce the texture specifications and meanwhile preserve the generating ability, we train our texture generator by fine-tuning on the pretrained LDM.

We train our model on the 3DBiCar dataset for its topology consistent mesh model and diversity of cartoon identities. Our training samples consist of a 3D mesh, a geometry related UV texture map and a corresponding description. Since the detail description is not supported, we first propose a multi-agent character captioning pipeline for generating detail caption of each 3D textured model.

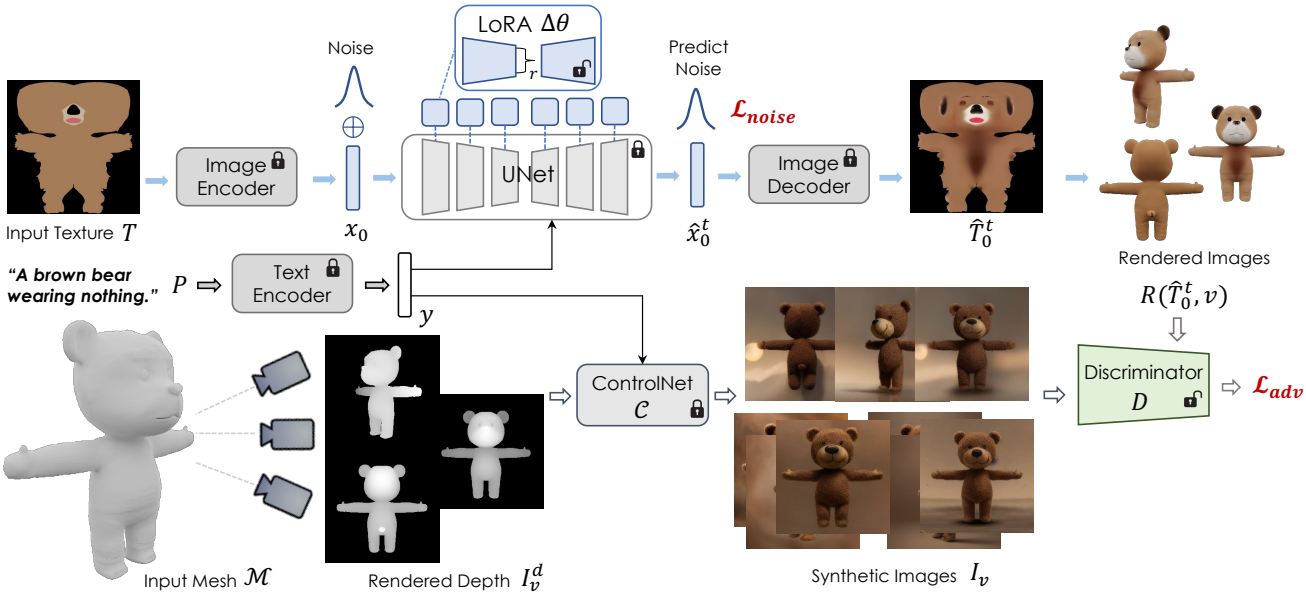

Figure 3. Overall framework for training texture generator. Our method takes a pair of data as input including a texture map T, corresponding text description P and mesh model $\mathcal{M}$. We finetune the low-rank adaptor $\Delta\theta$ for pretrained text-to-image diffusion model to generate high quality UV texture. In order to improve the quality and perceptual fidelity of synthetic textures, we introduce adversarial training to enhance the texture details. We leverage synthetic plausible images I_v conditioned by the rendered depth I_v^d generated by ControlNet $\mathcal{C}$ as a proxy to guide this adversarial training.

4.1. Multi-agent Character Captioning

In this section, we focus on building a text-texture paired data for training the texture generator. In order to obtain the detailed description of the 3D model, inspired with [57], we propose a 3D caption pipeline which focuses on identity and texture information. The illustration of the caption pipeline is shown on Fig 2. Specifically, for each model, we firstly render multi-view images for each textured model and use Visual Question Answering(VQA) model, BLIP2 [23], to obtain detailed corresponding descriptions.

Similar with [56], we hard-code the first question as: 1) *"Describe the image in detail"* to let BLIP2 generated a brief initial description of the image. Then, in order to obtain more information about the detail attribute of the color and cloth types, we start with another two specific questions to encourage the attention: 2) *"What is the color of this object?"*. 3) *"What is the object wearing?"*.

Furthermore, in order to enrich image captions and generate more informative descriptions, we integrate strong vision-language model, ChatGPT [36], for asking relevant questions according to the previous knowledge and progressively obtain more information. ChatGPT is prompted to ask follow-up questions to investigate more information about the image. Besides, in order to avoid the caption model for generating pose ore action-related information, we deign a head instruction of BLIP2 including: *"Answer given questions. Don't answer any contents about the pose or action of the object."*

At last, we use ChatGPT to summarize the descriptions across multi-views and result in the final caption. ChatGPT is able to merge the similar detail information in multi-views and remove the unlikely ones. We use the final caption as the detailed prompt of the 3D model and 3D texture for subsequent training.

4.2. Enhanced UV Texture Generation from Text

Now we use the prepared data for vivid and high-quality UV texture generation. Each pair of data includes a mesh model $\mathcal{M}$, a texture map $T \in \mathbb{R}^{H\times W\times 3}$ and a corresponding caption P. To ensure the generation of similar patterns or templates of the UV map within the dataset while preserving the generating capabilities, we customize specific parameters of pretrained text-to-image diffusion model. This customization allows us to leverage image knowledge as semantic priors in our texture generation process. We first start with a simple baseline that a parameter-efficient finetuning, Low-Rank Adaptation (LoRA), on the U-Net of the pretrained latent diffusion model (LDM) [42]. We encode the input texture T into latent x_0 and achieve diffusion process. The objective of the training is:

$$\mathcal{L}_{diff} = \mathbb{E}_{\epsilon,x_0,t}[||\epsilon - \epsilon_{\theta+\Delta\theta}(\sqrt{\alpha}_t x_0 + \sqrt{1-\alpha_t}\epsilon)||_2^2]. \quad (2)$$

$\Delta\theta$ denotes the tuned parameters, ϵ denotes the random noise map, $\epsilon_{\theta+\Delta\theta}(\cdot)$ is the predicted noise generated by denoiser integrated with LoRA adapter, α_t is the parameter of noise scheduler at timestamp t.

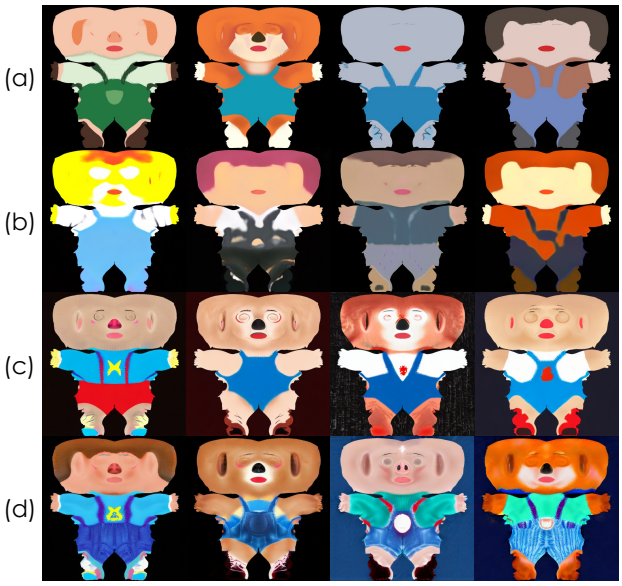

Figure 4. We showcase texture samples related with prompt *"wearing overall"* (a) from 3DBiCar [28] dataset; (b) generated by the texture generator from Rabit [28]; (c) generated by the simple finetuned LDM; (d) generated by the enhanced fintuned LDM.

After fine-tuning, we can infer the LDM and generate plausible texture map results related with the text instructions. However, simple fine-tuning on the collected dataset can only achieve UV-related texture map in the same domain with the dataset, result in limit concept and style variations. We show some synthetic results in Fig 4 related to the text "overall". We first show some selected texture maps from the dataset in (a), the selected results generated by the text generator of [28] in (b) and the synthetic results generated by the simple LDM in in (c). It is obvious that a simple finetuning of LDM tend to synthesis the structure of the overall without fine-grained details such as cloth wrinkles.

Therefore, bring the original texture domain to the realistic domain is necessary as it ensures the perceptual realism of the textured 3D model. To fix the domain issue and boost the quality of texture with fine-grained local details, we need a set of realistic texture image to represent the realistic distribution. However, it is hard to build or collect UV texture data that meets the requirements and related with the character geometry. Hence, we turn to create vivid synthetic images that looks like object renderings from different viewpoints.

Specifically, we apply a depth-guided image generator of ControlNet [55] to produce multi-view images guided by the rendered depth. Then we propose to impose an adversarial loss simultaneously when fine-tuning the parameters of the adapter. At each iteration, we randomly sample a camera view v from the pre-defined view set $\mathcal{V}$ and render the input mesh $\mathcal{M}$ to multi-view depth images. ControlNet receives the depth image I_v^d the text prompt P corresponding with the object, and, in response, synthesis high-quality images: $I_v = \mathcal{C}(I_v^d, y)$, where y denotes the text embedding of P. In our case, we set number of renderings for each 3D characters $|\mathcal{V}| = 8$. As for generated sample, we randomly sample a timestamp $t \in (0, 1000)$ and achieve diffusion process. Then we use the pretrained decoder to decode the denoised latent $\hat{x}_0^t = (x_t - \sqrt{1-\alpha_t}\epsilon_{\theta+\Delta\theta}(x_t, y, t))/\sqrt{\alpha_t}$ to the image $\hat{T}_0^t$. Then we use a differentiable mesh renderer $\mathcal{R}$ to render the textured mesh with texture $\hat{T}_0^t$ at view v. The render output is denoted as $\mathcal{R}(\hat{T}_0^t, v)$. And then we adopt adversary loss to make the rendered image $\mathcal{R}(\hat{T}_0^t, v)$ has the similar local structure and perceptual realism with the generated 2D images I_v at the same view. The objective of the adversarial training can be formulated as:

$$\mathcal{L}_{adv} = \mathbb{E}_{t,x_0,\epsilon}[logD(\mathcal{R}(\hat{T}_0^t, v)] + \mathbb{E}_{x_0}[log(1 - D(I_v))], \tag{3}$$

where the rendered image $\mathcal{R}(\hat{T}_0^t, v)$ is considered as a fake image while the output of the ControlNet is considered as the real sample. D is the adversarial discriminator that tries to maximize $\mathcal{L}_{adv}$. The boost texture results is shown on Fig 4(d). We can see that after the adversarial training, the network is able to generate more realistic texture details.

Texture seam fixing. When texturing the 3D model with synthetic UV texture image, we find that using image generative model would inevitably ignore the consistency at the seam of the 3D model and results in black seam artifacts. This might due to the reason that each processed texture data is agnostic to the whole perspective 3D knowledge. To help fix this issue , we first apply a Gaussian filter around the boundary part of the texture image, which will remove the "black seam" on the back of the model. However, this cannot solve the misalignment at the boundary. Therefore, we also conduct a simple image restoration technique on the back view of the model to mitigate the problem.

Specifically, for the generated texture map x_0, we render the textured mesh using the renderer $\mathcal{R}$ to obtain the rendered image T_v as seen from the back view. Then we apply a state-of-the-art image restoration method [51] to help make the rendered view perceptualy realistic without seam artifacts. Then we back project the I_v to the updated texture.

5. Experiments

5.1. Dataset

Our model is trained on 3DBiCar [28] dataset. 3DBiCar spans a wide range of 3D biped cartoon characters, containing 1,500 high-quality 3D models. The 3D cartoon characters have diverse identities and shape resulting in 15 character species, including *Human, Bear, Mouse, Cat, Tiger, Dog, Rabbit, Monkey, Elephant, Fox, Pig, Deer, Hippo, Cattle and Sheep*. All the 3D models are rigged and skinned by the predefined skeleton and skinning weight matrix, which supports further animation. Note that eyeball meshes and

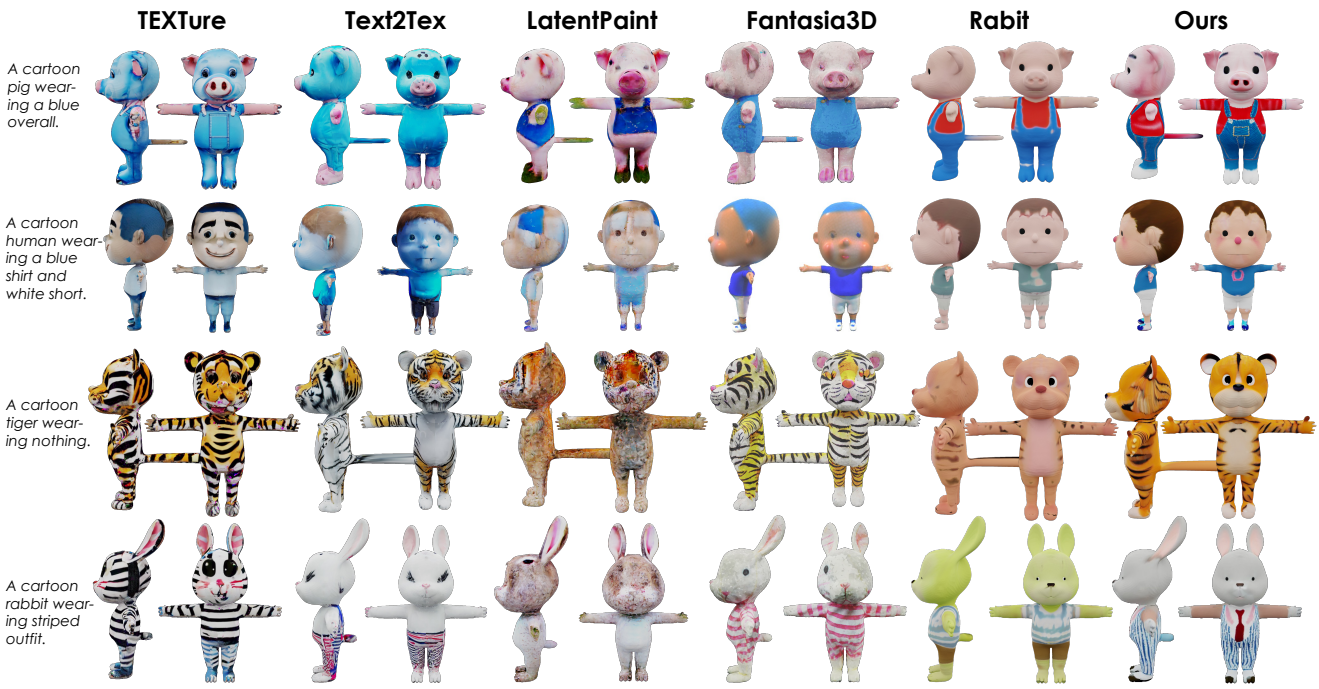

Figure 5. Qualitative comparison on the test prompt set with state-of-the-art shape texturing approaches, TEXTure [41], Text2Tex [7], LatentPaint [30], Fantasia3D [8] and Rabit [28], We show our results with high-quality and consistent texture faithful to the input prompt.

	CLIP↑	Time(min)↓	GPU(GB)↓
LatentPaint [30]	27.15	13.95	11.46
Fantasia3D [8]	29.20	21.55	12.42
Text2Tex [7]	28.81	14.35	20.31
TEXTure [41]	29.25	2.38	12.05
Rabit [28]	-	0.01	2.39
Ours	29.86	0.03	6.20

Table 1. Quantitative comparison on the test prompt set with other state-of-the-art texturing method. We also report the inference time for a single prompt and GPU memory to show our efficiency.

textures are extra modeled to support the facial expression in the future better. In our experiments, we use the default texture for eyeballs.

5.2. Implementation Details

Our texture generator is fine-tuned based on the cutting edge open source model Stable Diffusion [42] version 1.5. We inject LoRA into the projection matrices of query, key and value in all of the attention modules. We set the rank of the LoRA to 8. Then the modified forward pass of input x is: $h = W_0 x + B_{uv} A_{uv} x$, where $B_{uv} A_{uv}$ denote the parameters of adapter. We fine-tune the adapter using the AdamW [27] with a learning rate $1e-4$. For inference, we use classifier-free guidance with a guidance weight ω : $\hat{\epsilon}_\phi(x_t; y, t) = (1+\omega)\epsilon_\phi(x_t; y, t) - \omega\epsilon_\phi(x_t; t)$. In our experiments, we set $\omega = 7.5$. All the training and inference are performed on a single NVIDIA A100 GPU.

	FID↓	KID($\times 10^{-3}$)↓
Rabit [28]	42.55	6.37
Ours	35.25	5.25

Table 2. Quantitative comparison on the 3DBiCar dataset. Since results of other approaches have a large domain gap with the original texture dataset, so we only compare with Rabit [28] trained on the same dataset.

5.3. Comparison with State-of-the-art Approach

To the best of our knowledge, we are the first method focusing on texture creation in UV space of 3D biped cartoon model under text guidance. For far comparison, we build a test benchmark consisting of 300 test prompts, comprising all the 15 different species in dataset. For each species, we design 20 types of attributes about different combinations of cloth type and color. All the prompts follow the template: ***A cartoon [Species Name] wearing [Cloth Type].*** For example: *"A cartoon rabbit wearing blue shirt and white pants."* Then we select 15 mesh models for all species from the dataset as the base mesh and texture each 3D model using corresponding text prompts. For example, we apply the texture generated from *"A cartoon bear wearing suits."* on the *bear* mesh. For all baselines, we render 8 views of each textured object with white background using the same renderer setting from Blender [5] under resolution 1024*1024.

Baselines. We compare our method against two types of shape texturing approach: one is based on multi-view texture optimization, the other paints shape in a progres-

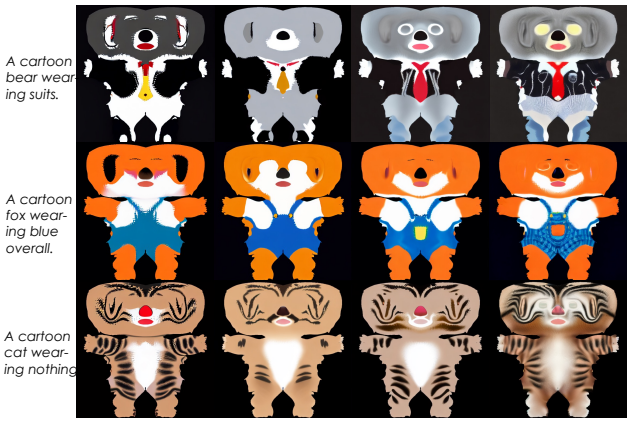

Figure 6. Ablation studies of different hyper-parameters and technical components. We visualize the synthetic results of models trained with different settings. (d) denotes our current setting.

sive manner. We first compare our approach with LatenPaint [30] and Fantasia3D [8], which optimize a 3D implicit scene based on the explicit mesh guided by text under multi-view SDS loss. For Fantasia3D, we only initialize the DMTet [44] based on the conditioned mesh model, and optimize the texture appearance under the correspondent text prompts. For painting methods, we compare with TEXTure [41] and Text2tex [7], which progressively generates partial textures across viewpoints and back-projects them to the texture space. Besides, we also compare the results from our texture generator with the StyleGAN2-based texture generator proposed by Rabit [28].

Qualitative comparisons. We compare the rendering results across several geometries textured from our approach against other baselines, as shown in Fig 5. We can see that our method is able to generated consistent texture to align faithfully with the conditioned text prompt. In contrast, painting-based methods like TEXTure and Text2Tex have noticeable seam artifacts when viewing the side and back sides of outputs. Optimizing-based methods can generate multi-view consistent texture. However, LatentPaint can hardly generate high-quality and text-related neural texture. While Fantasia3D demonstrates improved rendering results, there are still noticeable non-smooth artifacts present on the surfaces. We provide additional visualization of results using the texture generator from Rabit [28]. Since Rabit can only generate texture image unconditionally, so we randomly generate 100 texture maps and select relative results for visually comparison. The results show that the synthetic texture exhibits low-quality with indistinct structure. We conduct the user study to obtain the user's subjective evaluation of the fidelity and plausibility of the texture results. The detail can be found in *Supplementary Material*.

Quantitative comparisons. We evaluate the text-driven synthetic textures using average CLIP score to measure the alignment between texture image with the conditioned text prompts. The results is shown in Table 1. From the result we can see that our model achieves the best CLIP scores, indicating better text-texture alignment. We also report the run time for generating texture under a specific text guidance using the default hyper-parameters of each method on a single GPU. Notably, our method and Rabit are significantly faster than the optimization-based methods which indicates our efficiency. Besides, we also use the image quality and diversity metric Frechet Inception Distance (FID) [15] and Kernel Inception Distance (KID) [3] in Table 2. In our experiments, on 3DBiCar dataset, the real distribution comprises renders of the geometries with the same settings using their artist designed textures. Results show that our method achieves better score than the texture generator of Rabit in terms of both FID and KID.

5.4. Ablation Studies

We perform extensive ablation studies on different choices of hyper-parameters and the importance of the proposed adversarial learning scheme to investigate their effects on the final results. Specifically, we vary the rank of the LoRA adapter, exploring settings of 64, 32, and 8 training without adversarial loss. Then we investigate the effect of adversarial training for texture enhancement. The visualization results are presented in Figure 6, where the qualitative analyses unveil the influence of different settings on texture quality and diversity. According to the visualizations, it is evident that finetuning with a large rank introduces noticeable sawtooth artifacts. While reducing the rank mitigates this issue, it concurrently leads to textures with a low-poly and excessively smooth appearance. Lower ranks, such as 8, tend to yield more plausible semantic details. Adding adversarial training will help to enhance the fine-grained patterns in the texture output. Similar visualization results are also shown on the last two rows on the Fig 4.

6. Applications

Out of domain texture generation. Our method could enable realistic UV texture generation that highly faithful with the text instruction, and even support out of domain generation such as fashion icons or unreal humanoid characters from famous fiction or movies while retaining high recognition. We show some of results in Fig 7.

Prompt-based local editing. We also explore the controllability of our model as a prompt-based editing method in Fig 7. Simply using prompt-based editing can help to modify the texture according to the text while retaining other concepts. Such an editing capability makes the 3D texture creation with our model more controllable.

Stylized texture generation. Besides, we can achieve stylization for generated texture by injecting additional parameters from the other pretrained adapter $\mathcal{S}$ training on the styled image set. Then the modified forward pass of an in-

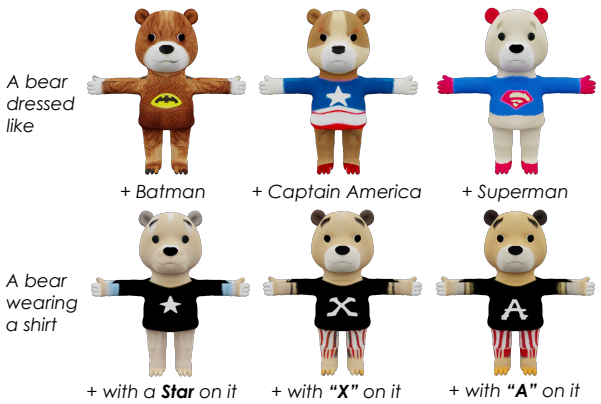

Figure 7. Make-It-Vivid enables out of domain generation about famous fictions and prompt-based local editing.

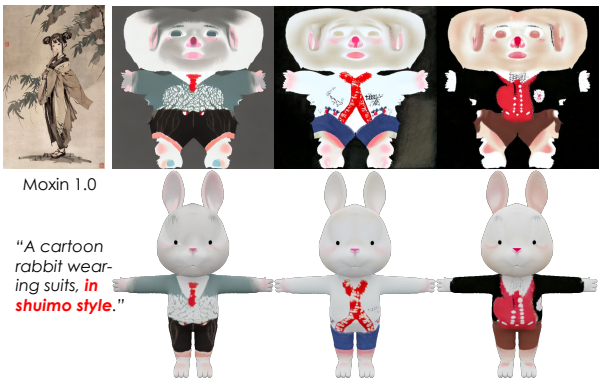

Figure 8. Make-It-Vivid enables stylized texturing. We show some synthetic results in shuimo style generated from our method injected with adapter MoXin 1.0 [33].

put x is: $h = W_0 x + B_{uv} A_{uv} x + w B_s A_s x$, where $B_s A_s$ denotes the parameters of $\mathcal{S}$. We set the balance weight $w = 0.5$. We show some samples generated by our model and a pretrained adapter MoXin1.0 [33] which is trained in a ink and wash painting dataset. We can see that after the stylization, the model is encouraged to generate plausible and stylized cloth types which takes large gap with original domain while preserve the original structure.

Textured characters production and animation. Our method aims to help users to create and customize vivid and plausible cartoon character efficiently. Therefore, we show the progressive generation system capable of creating textured animatable characters, driven by either text or video in Figure 9. Specifically, given a text prompt, we first employ the Large Language Model (LLM) [46] to process the text and extract three information including subject, texture and motion. For subject, we leverage a CLIP-based retrieval method to retrieve the shape with the nearest semantic in the dataset as the base geometry. Then we leverage our proposed texture model to design its appearance. To generated related motion according to the text, we directly apply a state-of-the-art text to motion model [14] to process the text and generate body rotation parameters. We then derive animated characters by applying the generated rotation parameters to the pre-defined joint points. Besides, we can also use video or other human interactions to drive or animate the created cartoon character.

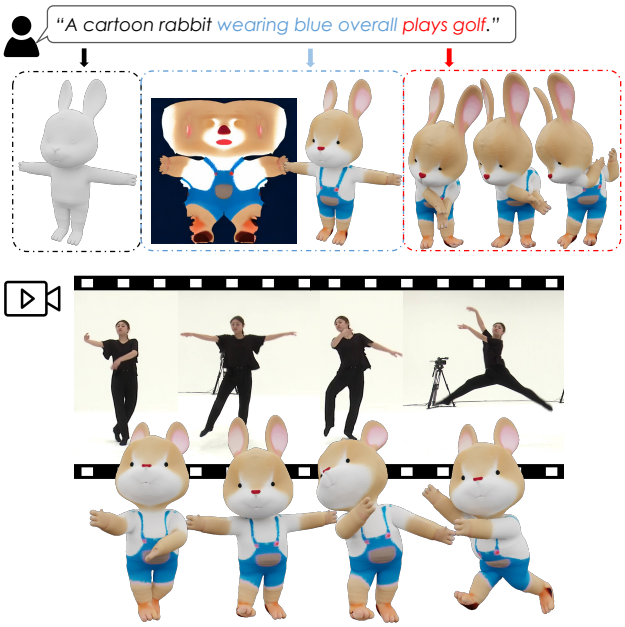

Figure 9. Make-It-Vivid supports efficient characters production and animation under text or video input.

7. Conclusion

We propose a novel text-guided texture generation in UV space for 3D biped cartoon characters, which enables to generate high-quality and semantic plausible UV textures. To accomplish the lack of high-fidelity data, we leverage priors from pretrained text-to-image model, which helps to generate texture map with template structure while preserving the natural knowledge. Furthermore, we propose an adversarial loss to shorten the domain gap between original dataset and realistic texture domain while training. Experiments show that our model can achieve efficient texture creation faithful with text input, supporting multiple stylization and local editing. Our approach can be easily applied to 3D character production and animation system, advance the 3D content creation.

8. Acknowledgement

The research is supported by National Key R&D Program of China (No. 2022ZD0161600), National Natural Science Foundation of China (72192821,62302297), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), Shanghai Science and Technology Commission (21511101200), Shanghai Sailing Program (22YF1420300, 23YF1410500) and Young Elite Scientists Sponsorship Program by CAST (2022QNRC001).

References

[1] Animal Crossing. https://animalcrossing.nintendo.com/. 1

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. 2

[3] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 7

[4] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 157–164. 2023. 3

[5] Blender. https://www.blender.org/. 6

[6] Tianshi Cao, Karsten Kreis, Sanja Fidler, Nicholas Sharp, and Kangxue Yin. Texfusion: Synthesizing 3d textures with text-guided image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4169–4181, 2023. 1, 2

[7] Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. *arXiv preprint arXiv:2303.11396*, 2023. 2, 6, 7

[8] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *arXiv preprint arXiv:2303.13873*, 2023. 1, 2, 6, 7

[9] DeepFloyd IF. https://www.deepfloyd.ai/deepfloyd-if. 2

[10] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 2

[11] Kentaro Fukamizu, Masaaki Kondo, and Ryuichi Sakamoto. Generation high resolution 3d model from natural language by generative adversarial network. *arXiv preprint arXiv:1901.07165*, 2019. 2

[12] Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos Zafeiriou. Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1155–1164, 2019. 2

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[14] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5152–5161, 2022. 8

[15] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 7

[16] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 867–876, 2022. 2

[17] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 2, 3

[18] Alexandros Lattas, Stylianos Moschoglou, Baris Gecer, Stylianos Ploumpis, Vasileios Triantafyllou, Abhijeet Ghosh, and Stefanos Zafeiriou. Avatarme: Realistically renderable 3d facial reconstruction" in-the-wild". In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 760–769, 2020. 2

[19] Alexandros Lattas, Stylianos Moschoglou, Stylianos Ploumpis, Baris Gecer, Jiankang Deng, and Stefanos Zafeiriou. Fitme: Deep photorealistic 3d morphable model avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8629–8640, 2023.

[20] Myunggi Lee, Wonwoong Cho, Moonheum Kim, David Inouye, and Nojun Kwak. Styleuv: Diverse and high-fidelity uv map generative model. *arXiv preprint arXiv:2011.12893*, 2020. 2

[21] Jiabao Lei, Yabin Zhang, Kui Jia, et al. Tango: Text-driven photorealistic and robust 3d stylization via lighting decomposition. *Advances in Neural Information Processing Systems*, 35:30923–30936, 2022. 1, 2

[22] Chenghao Li, Chaoning Zhang, Atish Waghwase, Lik-Hang Lee, Francois Rameau, Yang Yang, Sung-Ho Bae, and Choong Seon Hong. Generative ai meets 3d: A survey on text-to-3d in aigc era. *arXiv preprint arXiv:2305.06131*, 2023. 2

[23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 4

[24] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017. 3

[25] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 2

[26] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 3

[27] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[28] Zhongjin Luo, Shengcai Cai, Jinguo Dong, Ruibo Ming, Liangdong Qiu, Xiaohang Zhan, and Xiaoguang Han. Rabit: Parametric modeling of 3d biped cartoon characters

with a topological-consistent dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12825–12835, 2023. 1, 2, 3, 5, 6, 7

[29] Yiwei Ma, Xiaoqing Zhang, Xiaoshuai Sun, Jiayi Ji, Haowei Wang, Guannan Jiang, Weilin Zhuang, and Rongrong Ji. X-mesh: Towards fast and accurate text-driven 3d stylization via dynamic textual guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2749–2760, 2023. 1, 2

[30] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12663–12673, 2023. 1, 2, 6, 7

[31] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13492–13502, 2022. 2

[32] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 conference papers*, pages 1–8, 2022. 2

[33] MoXin. https://civitai.com/models/12597/moxin. 2, 8

[34] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 2

[35] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022. 2

[36] OpenAI. https://openai.com/blog/chatgpt. 4

[37] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2

[38] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2

[39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[40] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 2

[41] Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. Texture: Text-guided texturing of 3d shapes. *arXiv preprint arXiv:2302.01721*, 2023. 1, 2, 6, 7

[42] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 4, 6

[43] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 2

[44] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. *Advances in Neural Information Processing Systems*, 34:6087–6101, 2021. 2, 7

[45] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. *arXiv preprint arXiv:2303.14184*, 2023. 2

[46] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 8

[47] Christina Tsalicoglou, Fabian Manhardt, Alessio Tonioni, Michael Niemeyer, and Federico Tombari. Textmesh: Generation of realistic 3d meshes from text prompts. *arXiv preprint arXiv:2304.12439*, 2023. 2

[48] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12619–12629, 2023. 2

[49] Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, et al. Rodin: A generative model for sculpting 3d digital avatars using diffusion. *CVPR*, 2023. 2

[50] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023. 2

[51] Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. Diffir: Efficient diffusion model for image restoration. *arXiv preprint arXiv:2303.09472*, 2023. 5

[52] Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Zhengzhe Liu, and Xiaojuan Qi. Texture generation on 3d meshes with point-uv diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4206–4216, 2023. 2

[53] Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. Styleswin: Transformer-based gan for high-resolution image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11304–11314, 2022. 2

[54] Longwen Zhang, Qiwei Qiu, Hongyang Lin, Qixuan Zhang, Cheng Shi, Wei Yang, Ye Shi, Sibei Yang, Lan Xu, and Jingyi Yu. Dreamface: Progressive generation of ani-

matable 3d faces under text guidance. *arXiv preprint arXiv:2304.03117*, 2023. 2

[55] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 5

[56] Deyao Zhu, Jun Chen, Kilichbek Haydarov, Xiaoqian Shen, Wenxuan Zhang, and Mohamed Elhoseiny. Chatgpt asks, blip-2 answers: Automatic questioning towards enriched visual descriptions. *arXiv preprint arXiv:2303.06594*, 2023. 4

[57] Deyao Zhu, Jun Chen, Kilichbek Haydarov, Xiaoqian Shen, Wenxuan Zhang, and Mohamed Elhoseiny. Chatgpt asks, blip-2 answers: Automatic questioning towards enriched visual descriptions. *arXiv preprint arXiv:2303.06594*, 2023. 4

[58] Zootopia. https://en.wikipedia.org/wiki/Zootopia. 1