[Supplementary Material] TexGen: Text-Guided 3D Texture Generation with Multi-view Sampling and Resampling

Dong Huo[1,3*], Zixin Guo[2], Xinxin Zuo[3], Zhihao Shi[3], Juwei Lu[3], Peng Dai[3], Songcen Xu[3], Li Cheng[1], and Yee-Hong Yang[1]

[1] University of Alberta, Canada
{dhuo, lcheng5}@ualberta.ca, yang@cs.ualberta.ca
[2] University of Toronto, Canada
zixin.guo@mail.utoronto.ca
[3] Huawei Noah's Ark Lab
{xinxin.zuo1, zhihao.shi, juwei.lu, peng.dai, xusongcen}@huawei.com

1 Algorithm Details

To better illustrate the working flow of our proposed method, we present the detailed algorithm in Alg. 1

2 Derivation of Eq. 12

As discussed in Eq. 12 of Sec. 3.3 in the main paper, we apply the classifier-free guidance (CFG) on noise estimation with two conditions: the textual prompt c and the intermediate texture map $\hat{U}_t^i$. The original text guided diffusion model targets at learning $P(x_t|c)$ where x_t denotes the noisy latent feature at time step t. Now we extend the target of the original diffusion model to $P(x_t^i|c, \hat{U}_t^N)$, which has an additional condition $\hat{U}_t^N$ to constrain the generated x_t^i to be view-consistent. We assume $P(c|x_t^i, \hat{U}_t^N) = P(c|x_t^i)$. Following Bayes' theorem, $P(x_t^i|c, \hat{U}_t^N)$ can be reformulated as

$$P(x_t^i|c, \hat{U}_t^N) = \frac{P(x_t^i)P(c|x_t^i)P(\hat{U}_t^N|x_t^i)}{P(c, \hat{U}_t^N)}. \tag{13}$$

By taking logarithm on both sides of the above equation, we get

$$\begin{aligned}\log(P(x_t^i|c, \hat{U}_t^N)) =& \log(P(x_t^i)) + \log(P(c|x_t^i)) \\ &+ \log(P(\hat{U}_t^N|x_t^i)) - log(P(c, \hat{U}_t^N)).\end{aligned} \tag{14}$$

As mentioned in [4], estimating $\epsilon_m(x_t^i)$ is related to predicting the score function $s_m(x_t^i)$ of the approximate marginal distribution $P(x_t^i|c, \hat{U}_t^N)$, which can be formulated as:

$$s_m(x_t^i) = \nabla_{x_t^i} \log(P(x_t^i|c, \hat{U}_t^N)), \tag{15}$$

* Work done during an internship at Huawei Noah's Ark Lab

Algorithm 1: Text-Guided 3D Texture Generation with Multi-view Sampling and Resampling

Input: A 3D Mesh
A textual prompt c
A set of viewpoints v_i, $i \in \{1, \dots, N\}$
Number of denoising step T
VAE encoder $\mathcal{E}$ and decoder $\mathcal{D}$
Depth-conditioned ControlNet $Unet_\theta$

Output: Generated texture map $\hat{U}_1^N$

Randomly initialize $x_T^i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $i \in \{1, \dots, N\}$
for $t = T, ..., 1$ **do**
- **Attention-Guided Multi-view Sampling:**
- **for** $i = 1, \dots, N$ **do**
  - Substitute the Key and Value features for viewpoint i with those from reference view to calculate $\epsilon_\theta(x_t^i)$ by Eq. 6 and Eq. 7
  - Obtain the $\hat{x}_0^i(x_t^i)$ with x_t^i and $\epsilon_\theta(x_t^i)$ by Eq. 2
  - Decode the $\hat{x}_0^i(x_t^i)$ to obtain I_t^i in RGB space by Eq. 3
  - Inverse render the I_t^i to obtain the partial texture map $\hat{U}_t^i$
  - **if** $i < N$ **then**
    - Render and encode $\hat{U}_t^i$ to obtain G_t^{i+1} by Eq. 4
    - Update x_t^{i+1} with G_t^{i+1} and observation mask $\mathcal{M}^{i+1}$ by Eq. 5
  - **end**
- **end**
- **Text&Texture-Guided Resampling:**
- **for** $i = 1, \dots, N$ **do**
  - Calculate the $\hat{\epsilon}_{tex}(x_t^i)$ with $\hat{U}_t^N$ by Eq. 8
  - Obtain the texture-conditioned noise estimation $\epsilon_{tex}(x_t^i|\hat{U}_t^N)$ by Eq. 11
  - Combine the texture-conditioned noise estimation $\epsilon_{tex}(x_t^i|\hat{U}_t^N)$, text-conditioned noise estimation $\epsilon_\theta(x_t^i|c)$, and unconditioned noise estimation $\epsilon_\theta(x_t^i|\varnothing)$ to calculate the final noise estimation $\epsilon_m(x_t^i)$ by Eq. 12
  - **if** $t > 1$ **then**
    - Substitute $\epsilon_\theta(x_t^i)$ with $\epsilon_m(x_t^i)$ in Eq. 1 and Eq. 2 to calculate the x_{t-1}^i for the next denoising step
  - **end**
- **end**

end

$$\epsilon_m(x_t^i) = -\sigma_t s_m(x_t^i), \tag{16}$$

where σ_t is the standard deviation of the latent noise parameterized by denoising step t. The score function $\nabla_{x_t^i} \log(P(x_t^i|c, \hat{U}_t^N))$ can be further derived from Eq. 14 as:

$$\begin{aligned} \nabla_{x_t^i} \log(P(x_t^i|c, \hat{U}_t^N)) =& \nabla_{x_t^i} \log(P(x_t^i)) + \nabla_{x_t^i} \log(P(c|x_t^i)) \\ &+ \nabla_{x_t^i} \log(P(\hat{U}_t^N|x_t^i)), \end{aligned} \tag{17}$$

with

$$\nabla_{x_t^i} \log(P(c|x_t^i)) = \nabla_{x_t^i} \log(P(x_t^i|c)) - \nabla_{x_t^i} \log(P(x_t^i)), \tag{18}$$

$$\nabla_{x_t^i} \log(P(\hat{U}_t^N|x_t^i)) = \nabla_{x_t^i} \log(P(x_t^i|\hat{U}_t^N)) - \nabla_{x_t^i} \log(P(x_t^i)), \tag{19}$$

which correspond to the terms in our multi-conditioned CFG in Eq. 12 as:

$$\epsilon_\theta(x_t^i|\varnothing) = -\sigma_t \nabla_{x_t^i} \log(P(x_t^i)), \tag{20}$$

$$\epsilon_\theta(x_t^i|c) - \epsilon_\theta(x_t^i|\varnothing) = -\sigma_t(\nabla_{x_t^i} \log(P(x_t^i|c)) - \nabla_{x_t^i} \log(P(x_t^i))), \tag{21}$$

$$\epsilon_{tex}(x_t^i|\hat{U}_t^N) - \epsilon_\theta(x_t^i|\varnothing) = -\sigma_t(\nabla_{x_t^i} \log(P(x_t^i|\hat{U}_t^N)) - \nabla_{x_t^i} \log(P(x_t^i))). \tag{22}$$

Following CFG [3], we apply two guidance scales ω_1 and ω_2 on two guidance terms. Finally, we have the multi-conditioned CFG as:

$$\begin{aligned} \epsilon_m(x_t^i) =& \epsilon_\theta(x_t^i|\varnothing) \\ &+ \omega_1(\epsilon_\theta(x_t^i|c) - \epsilon_\theta(x_t^i|\varnothing)) \\ &+ \omega_2(\epsilon_{tex}(x_t^i|\hat{U}_t^N) - \epsilon_\theta(x_t^i|\varnothing)). \end{aligned} \tag{23}$$

3 Summary of Symbols

In Tab. 4, we summarize all symbols that are mentioned in the main paper with corresponding explanations.

4 Additional Experiments

4.1 Inference Time

In Tab. 5, we compare the inference time of our proposed method with that of baseline methods on a single NVIDIA Tesla V100 GPU with 32GB memory. Our runtime 30.83 minutes = ((VAE encoding and decoding 2.63s + Key and Value features substitution 0.55s + inverse rendering 1.49s + denoising sampling 0.47s)×9 views)×40 denoising steps. The decoding of latent features and encoding of a rendered RGB texture for each view at each denoising step increased the computation cost, as well as the differentiable inverse rendering for view assembling.

Table 4: All symbols and corresponding explanations

Symbols	Explanations
t	denosing step
T	total number of denosing step
i	index of a viewpoint
N	number of viewpoints
α_t	total noise variance parameterized via denoising step t
x_t^i	noisy latent feature of view i at denoising step t
x_t^{ref}	noisy latent feature of reference view at denoising step t
$\hat{x}_0^i(x_t^i)$	denoised observation of x_t^i
$\hat{U}(x_t^{1\cdots i}), i < N$	assembled noise-free partial texture map from view 1 to i
$\hat{U}(x_t^{1\cdots N})$	assembled noise-free complete texture map from all views
$\hat{U}_t^i$	abbreviation of $\hat{U}(x_t^{1\cdots i})$
$\hat{U}_t^N$	abbreviation of $\hat{U}(x_t^{1\cdots N})$
$\hat{U}_1^N$	final generated texture map
ω	user-specified weight for CFG
ω_1	user-specified weight for multi-conditional CFG
ω_2	user-specified weight for multi-conditional CFG
c	text prompt
$\varnothing$	null-text prompt
$\mathcal{D}$	VAE decoder of the pre-trained stable diffusion
$\mathcal{E}$	VAE encoder of the pre-trained stable diffusion
I_t^i	RGB image decoded from $\hat{x}_0^i(x_t^i)$
$Render^{i+1}(\hat{U}_t^i)$	render of partial texture map $\hat{U}_t^i$ at view $i+1$
$Render^{i}(\hat{U}_t^N)$	render of complete texture map $\hat{U}_t^N$ at view i
G_t^{i+1}	encoding of $Render^{i+1}(\hat{U}_t^i)$
$\mathcal{M}^{i+1}$	mask of regions observed for the first time at view $i+1$
$Unet_\theta$	Unet of stable diffusion
Q_t^{ref}	Query features from the self-attention module of the reference view
K_t^{ref}	Key features from the self-attention module of the reference view
V_t^{ref}	Value features from the self-attention module of the reference view
$\epsilon_\theta(x_t^i)$	estimated noise from x_t^i using the pre-trained diffusion model
$\epsilon_\theta(x_t^i \vert c)$	text-conditioned noise estimation
$\epsilon_\theta(x_t^i \vert \varnothing)$	unconditioned noise estimation
$\epsilon_{tex}(x_t^i \vert \hat{U}_t^N)$	texture-conditioned noise estimation
$\epsilon_\theta(x_t^i)$	linear combination of $\epsilon_\theta(x_t^i \vert \varnothing)$ and $\epsilon_\theta(x_t^i \vert c)$ based on CFG
$\epsilon_{tex}(x_t^i)$	linear combination of $\epsilon_\theta(x_t^i \vert \varnothing)$ and $\epsilon_{tex}(x_t^i \vert \hat{U}_t^N)$ based on CFG
$\hat{\epsilon}_{tex}(x_t^i)$	estimation of $\epsilon_{tex}(x_t^i)$ from the render of $\hat{U}_t^N$
$\epsilon_m(x_t^i)$	multi-conditioned noise estimation
ϵ	random Gaussian noise

Table 5: Inference time of compared methods using images with resolution of 512×512 on a single NVIDIA Tesla V100 GPU.

Methods	Inference Time (minutes) ↓
TEXTure	3.94
Text2Tex	20.64
Fantasia3D	109.67
ProlificDreamer	483.92
Ours	30.83

4.2 More Qualitative Evaluations

More qualitative evaluations are shown in Fig. 10, Fig. 11, and Fig. 12.

5 More Ablation Studies

We demonstrate the impact of the reference view in Fig. 14. The attention guidance from the reference view could maintain a high-level semantic similarity instead of pixel-wise consistency, therefore the choice of the reference view only impact the style of the generated texture.

6 User Study Details

We develop a WIX-based web application for the user study. As shown in Fig. 13, for each video pair, participants are required to choose the video that best illustrates the given textual prompt with the highest quality. They should then click the rounded check-box below the selected video and proceed to the next video pair. Finally, we determine the user preferences by counting all user selections.

7 Data Description

We present the details of our collected data in Tab. 6 and Tab. 7 with corresponding textual prompts.

Fig. 10: More texture generation results of our proposed method.

Fig. 11: Visual comparison of our proposed method against TEXTure [5] and Text2Tex [1].

Fig. 12: Visual comparison of our proposed method against Fantasia3D [2] and ProlificDreamer [6].

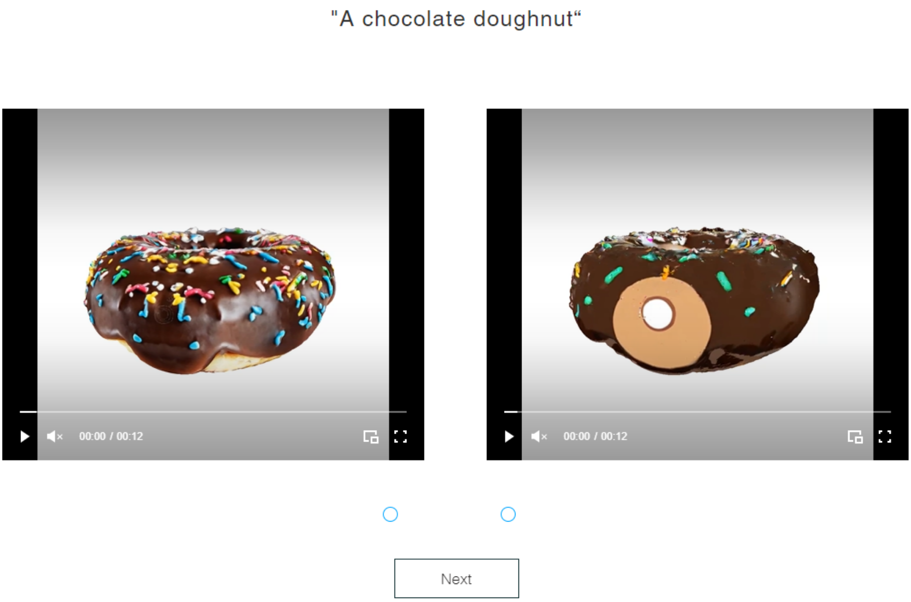

Fig. 13: Screenshot of the user study web application

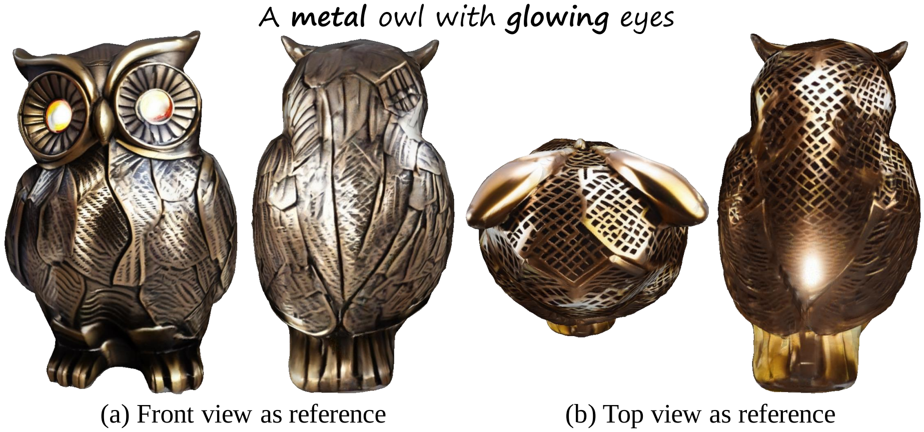

Fig. 14: The impact of reference views.

Table 6: Description of 3D Meshes in our collected data.

Object	Source	Description	Textual Prompts
eb219212147f4d84b88f8e103af8ea10	Objaverse	frog	"A robotic frog" "A green frog"
a8813ea1e0ce47ab97a416637a7520d7	Objaverse	helmet	"A Mandalorian helmet in silver" "A black helmet"
e0417d1e05984727a50f9ab1451d162d	Objaverse	lantern	"A stone lantern" "A medieval lantern"
9fa2da2c42234b58896e8d23393cac24	Objaverse	backpack	"A backpack in ironman style" "A backpack in spiderman style" "A 3D backpack"
a51751c9989940e592eb61be41ee35cc	Objaverse	baby owl	"A baby owl with fluffy wings" "A toy owl"
f73e2e1c8ad241ff859aca7e032ec262	Objaverse	lion	"A cute 3D cartoon lion with brown hair" "A marble lion"
91c5283b27c74583900d5e26e2fcd086	Objaverse	mug	"A wooden mug surrounded by silver rings" "A mug with cloud"
b6db59bd7f10424eae54c71d19663a65	Objaverse	car	"A next gen nascar in red" "A next gen nascar"
a2832b845e4e4edd9d439342cf4fd590	Objaverse	wolf	"Statue of a wolf" "A white wolf"
b19ef2650b4347348710eb6364ca90bd	Objaverse	penguin	"A black penguin" "A penguin covered by a blue sweater"
bd384d46514548cf8c4202f1ae6ea551	Objaverse	refrigerator	"A wooden refrigerator" "A high tech refrigerator"
f1aa479977a74a608d362679ed5ca721	Objaverse	piano	"A medieval piano" "A piano with flowers"
4c4690ba918f477b829990dd2e960c21	Objaverse	lion	"A golden lion" "A cyber punk lion"
f87caf6ac5a445ccad1a97653688e16e	Objaverse	dresser	"A wooden dresser" "A marble dresser"
f15298421b3d4e0fab4c43863a7e72fd	Objaverse	shark	"A deep ocean shark" "A dark blue shark"
d4c560493a0846c5943f3aeea58acb72	Objaverse	soccer ball	"A soccer ball in black and white" "A stone soccer ball"
c6509a8fe1f44a5eac8aebe12be2699e	Objaverse	tiger	"A tiger walking on the grass" "A plastic toy tiger"
fa2c41a7a6c84fcb871a24016fa9a932	Objaverse	doughnut	"A chocolate doughnut" "An icecream doughnut"
f05b0c2f9bcf41cea188a4b4c848068a	Objaverse	fireplug	"A fireplug, red and yellow" "A fireplug with yellow top"
bff537fb09b641c59b2ad123da0ca3dc	Objaverse	turtle	"A metal turtle with red eyes" "A sea turtle"
d726514a97f74f168b104fd6ba538331	Objaverse	vase	"An ancient vase" "A painted vase"
01ab0842feb1448bb18e8c7b85326d11	Objaverse	pottery	"An antique pottery" "A pottery with flowers"
f2d31eb0ddac4d21944df7dcc4af6d28	Objaverse	vending machine	"A coca cola vending machine" "A silver vending machine"
fc9cc06615084298b4c0c0a02244f356	Objaverse	piano	"A medieval piano" "A piano with flowers"
7adc9c74b75e4860b0a51c850bde9957	Objaverse	dress	"A princess dress" "A dress with spider patterns"
2fc0fc6ebe564a249c4617e6b3e6da93	Objaverse	fireplace	"A brick fireplace" "A stone fireplace"
14b8ae60eae240ff8bf1abdf9af5e49c	Objaverse	refrigerator	"A wooden refrigerator" "A high tech refrigerator"
62897c52e967469c85df9c6abdd09d16	Objaverse	doll	"A doll with yellow hairs" "A spiderman doll"
6f5480698a7a43c7a8c0a8b1e295e4a0	Objaverse	pumpkin	"A pumpkin with red eyes" "A Halloween pumpkin"

Table 7: Description of 3D Meshes in our collected data.

Object	Source	Description	Textual Prompts
e1f96691aaf648b885d927f5c3f5be61	Objaverse	apple	"A red apple" "An oil painted apple"
8a60954eccad433e987bbcafc7657140	Objaverse	armor	"A medieval armor" "A Japanese armor"
f98c5ee54c4a48f8b5eafd35a81dde4d	Objaverse	owl	"A metal owl with glowing eyes" "A wooden owl"
fadefc1eee3246a189f6b79c7c671343	Objaverse	lion	"A lion looking forward" "Statue of a lion"
9a0c52d350634e419aaf0eea1e67d9da	Objaverse	knight	"A golden knight" "A silver knight"
0db114d7753344d6825aa4f21ec56db9	Objaverse	crate	"A wooden crate" "A bronze crate"
72826cd5c17a42798a8e8e36c05c5035	Objaverse	clock	"A medieval clock" "A electric clock"
ac5df73de2c54239833643423a152592	Objaverse	dresser	"A wooden dresser" "A marble dresser"
90009fa6fa0b4d4bb1a1203431954097	Objaverse	keg	"A metal keg in silver" "A wooden keg"
b26a53419075442ca284cdf1d5541765	Objaverse	monitor	"A mac monitor" "An ironman monitor"
f75caead1dc1474195eb32a7f4c71117	Objaverse	control	"A game controller with black buttons on the top" "A PS5 controller"
edbeb81ef32645cea8bef89338f7e213	Objaverse	telephone	"A telephone with golden dials" "A classic telephone"
Napoleon ler	ThreeDScans	statue	"A high quality color photo of Tom Cruise" "A high quality color photo of Benedict Cumberbatch" "A high quality color photo of Robert Downey Jr."
Plastic Dragon	ThreeDScans	statue	"Cartoon dragon, red and green" "A 3D dragon"
Francois	ThreeDScans	statue	"Spiderman with white hairs" "A boy in suits"
Provost	ThreeDScans	statue	"Portrait of Provost, oil paint" "A statue of Provost"

References

1. Chen, D.Z., Siddiqui, Y., Lee, H.Y., Tulyakov, S., Nießner, M.: Text2tex: Text-driven texture synthesis via diffusion models. arXiv preprint arXiv:2303.11396 (2023)
2. Chen, R., Chen, Y., Jiao, N., Jia, K.: Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. arXiv preprint arXiv:2303.13873 (2023)
3. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
4. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022)
5. Richardson, E., Metzer, G., Alaluf, Y., Giryes, R., Cohen-Or, D.: Texture: Text-guided texturing of 3d shapes. arXiv preprint arXiv:2302.01721 (2023)
6. Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., Zhu, J.: Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. arXiv preprint arXiv:2305.16213 (2023)