SceneTex: High-Quality Texture Synthesis for Indoor Scenes via Diffusion Priors Supplementary Material

Dave Zhenyu Chen[1] Haoxuan Li[1] Hsin-Ying Lee[2] Sergey Tulyakov[2] Matthias Nießner[1]
[1]Technical University of Munich [2]Snap Research
https://daveredrum.github.io/SceneTex/

In this supplementary material, we present additional results and analysis in Sec. 1 to show the effectiveness of the proposed texture synthesis method. We also provide details of the user study in Sec. 2. For reproducibility, we produce the architecture details of the proposed Cross-attention Texture Decoder in Sec. 3, and the hash encoding [3] configurations of the multiresolution texture in Sec. 4.

1. Additional Qualitative Results

To further showcase the effectiveness of the proposed method, we present more texture synthesis results on 3D-FRONT dataset [2].

Realistic indoor scene decorations. Our method is capable of generating high-quality realistic scene appearance for most common indoor decoration styles, such as "Scandinavian" and "luxury", as shown in Fig. 2. Additionally, our method can move beyond common indoor styles to more challenging ones, such as "Lego" and "Game of Throne", as shown in Fig. 3. It demonstrates the flexibility of our method for synthesizing high-quality texture while loyally reflecting iconic properties in the input prompts. We believe that those synthesis results indicate a great potential to stylize more high-quality 3D textures for indoor scene.

Stylizing the same scene. In Fig. 4, we show that our method can generate different texturing results on the same objects. Here, we use the same prompt template "a ⟨STYLE⟩ living room" with 4 different styles ("Bohemian", "Baroque", "French country", and "Japanese"). All our textures are highly detailed and loyal to the style, demonstrating a great potential to generate diverse and various 3D scenes appearances.

2. User Study Details

We develop a Django-based web application for the user study. In Fig. 5, we show the interface for the questionnaire. We randomly select textured scenes from each baseline and our method to form a batch of 8 samples. To better visualize the textured scene, we render multi-view images for those scenes from 10 different viewpoints. After presenting the 10 rendered images, we ask the users to rate the appearance of the scenes from 1 to 5 in terms of aesthetics, realism, smoothness, etc. We also ask the users to measure how well the scene appearances match the given descriptions. To avoid biases and cheating in this user study, we shuffle scenes so that there is no positional hint of our method. In the end, we gather 100 responses from 75 participants to calculate the user study results.

Parameter	Value
Number of levels	20
Hash table size	2^{24}
Number of feature dimensions	4
Min. resolution	16^2
Max. resolution	4096^2

Table 1. **Multiresolution texture configuration.** We configure the hash encoding parameters following iNGP [3].

3. Cross-attention Texture Decoder

We implemented a neural texture decoder with cross-attention mechanism to strengthen the style-consistency within each object in the scene. As Fig. 1 shows, we map the UV embeddings with 2-layer MLPs in a 4-head cross-attention module implemented with Flash Attention v2 [1]. A skip connection is applied between the original rendering UV embeddings and the output features as the final instance-aware UV embeddings. The final RGB renderings are produced by a 4-layer MLP with hidden size 256.

4. Multiresolution Texture Configuration

In Tab. 1, we illustrate the implementation details of our multiresolution texture for reproducibility, following the configuration in iNGP [3]. Here, we use a deep hierarchy with 20 layers of texture features to learn scene appearance

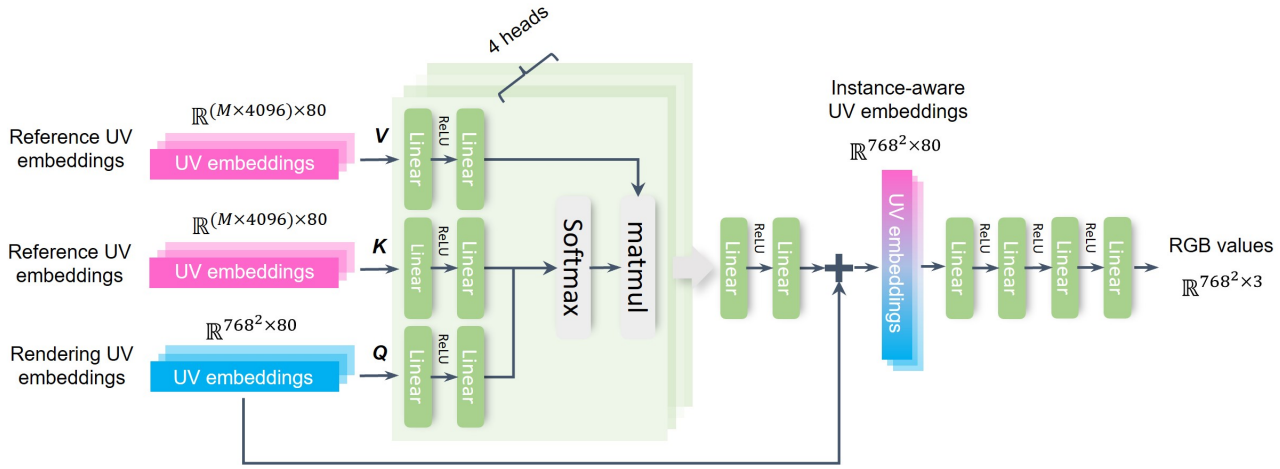

Figure 1. **Architecture details for Cross-attention Texture Decoder.** For the target 768×768 image, we directly obtain the rendering UV embeddings in shape $\mathbb{R}^{768^2\times80}$. Then, for the current scene with M objects, we query the reference UV embeddings in shape $\mathbb{R}^{(M\times4096)\times80}$ of the pre-sampled $4,096$ positions for each object. We apply 2-layer MLPs to map the UV embeddings to the Key, Value, and Query for the cross-attention module. We use a 4-head cross-attention module implemented with Flash Attention v2 [1]. The output features are mapped with a 2-layer MLP, and added to the original rendering UV embeddings as the final instance-aware UV embeddings. Finally, a 4-layer MLP predicts the RGB renderings of the current viewpoint.

at various scales. This ensures our synthesis architecture to faithfully encode detailed texture information irrespective of the distance between the viewpoint and mesh surface. To reduce possible hash collision in UV space, we choose a comparatively big hash table size of 2^{24}. Our multiresolution texture encodes texture features from 16×16 to $4,096 \times 4,096$.

References

[1] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023. 1, 2

[2] Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, et al. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10933–10942, 2021. 1

[3] 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. 1

Figure 2. **Synthesized scene textures with modern styles.** Our method is capable of generating high-quality scene appearance for common indoor decoration styles. Ceilings and back-facing walls are excluded for better visualizations. Images best viewed in color.

Figure 3. **Synthesized scene textures with creative styles.** Our method can also generate textures for challenging and creative scene styles. Ceilings and back-facing walls are excluded for better visualizations. Images best viewed in color.

Figure 4. **Synthesized textures for 3D-FRONT scenes.** Our method generates different textures for the same input scene. We use the prompt template "a ⟨STYLE⟩ living room" with 4 different styles for texture generation.

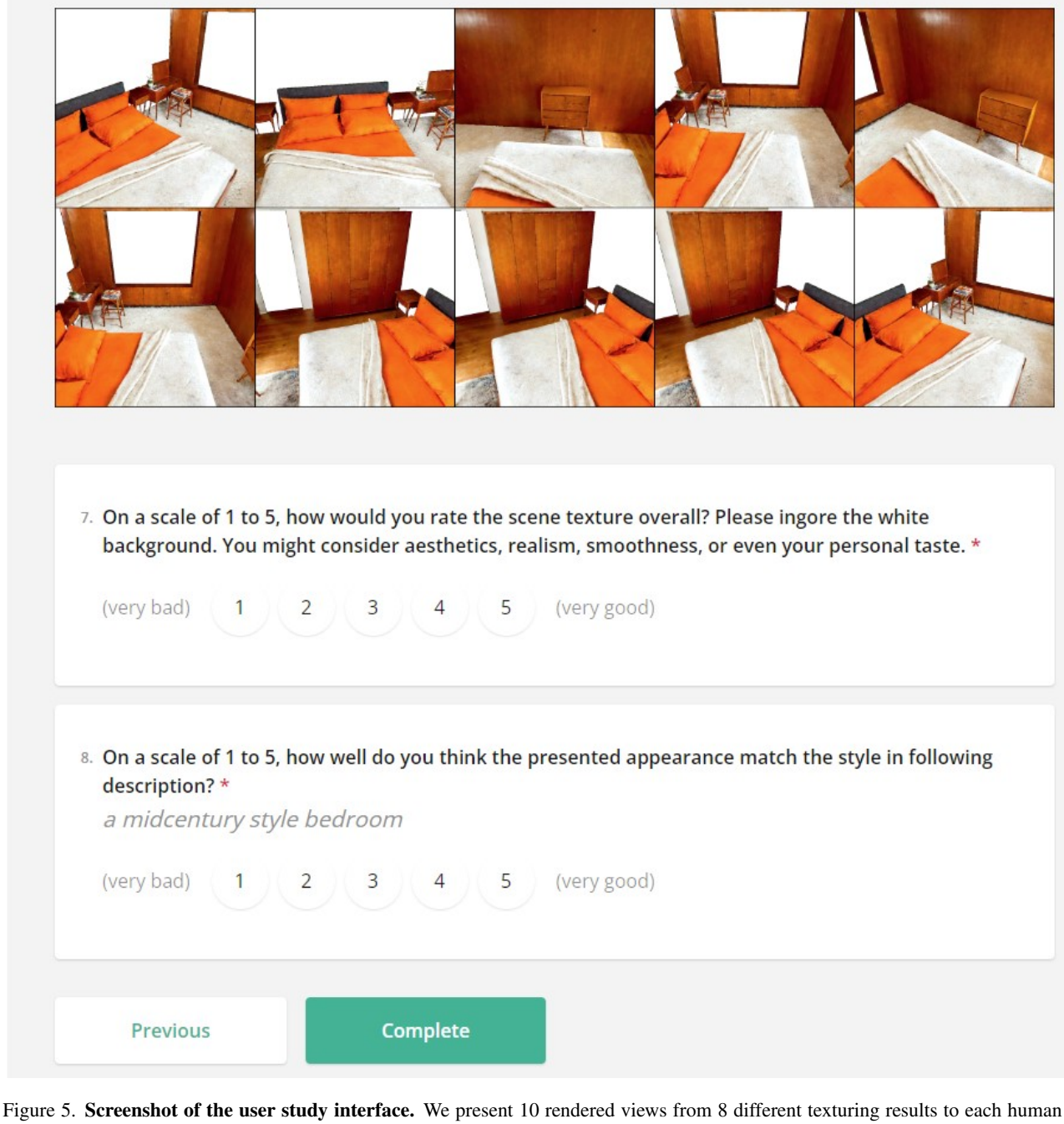

Figure 5. **Screenshot of the user study interface.** We present 10 rendered views from 8 different texturing results to each human user, and ask them to rate the appearance and the similarity with input prompts on a scale of 1-5.