NaTex: Seamless Texture Generation as Latent Color Diffusion Supplementary Material

Zeqiang Lai[1,2⋆] , Yunfei Zhao[2⋆] , Zibo Zhao[2] , Xin Yang[2]
Xin Huang[2] , Jingwei Huang[2] , Xiangyu Yue[1‡] , Chunchao Guo[2‡]
[1]MMLab, CUHK [2]Tencent Hunyuan
https://natex-ldm.github.io

A. Implementation Details

Training Details. To validate the proposed method, we train a color VAE with 300M parameters and a color DiT with 1.9B parameters using a flow-matching objective. The VAE is trained with a maximum of 6144 tokens, with token scaling during inference. For DiT training, we set the batch size to 256 and use a constant learning rate scheduler with a linear warm-up for the first 500 steps. The learning rate starts at 1×10^{-4} and decays to 1×10^{-5} thereafter. The illumination-invariant loss is introduced once pretraining converges, with a weight of 5. We adopt classifier-free guidance [1] by replacing conditioning embeddings with zero embeddings at a 10% probability during training. Unless otherwise stated, all results in this paper are obtained with 5 diffusion steps and a guidance scale of 2. The illumination-invariant loss is introduced once pretraining converges, with its weight set to 5.

Data Preparation. We use Blender to sample uniform color point clouds from raw meshes. For the input images, we render 24 views uniformly around the object, with random elevation angles in the range of 45° to -30°. We also randomly select from various illumination environments, including point lights, area lights, and HDRI maps.

Training & Inference Cost. We trained our model using 64 GPUs on a curated dataset of approximately 150k high-quality textured meshes. The training process lasted roughly one week. For our 2B parameter model, the peak memory consumption is approximately 24GB.The inference process for a non-distilled model (15 steps) takes roughly 100s for the DiT and 15s for the VAE, totaling approximately two minutes per case.

Network Architecture. We employ a Flux-like architecture, consisting of 12 double blocks and 24 single blocks. The hidden dimension is 1536, resulting in a total model size of 1.9B parameters.

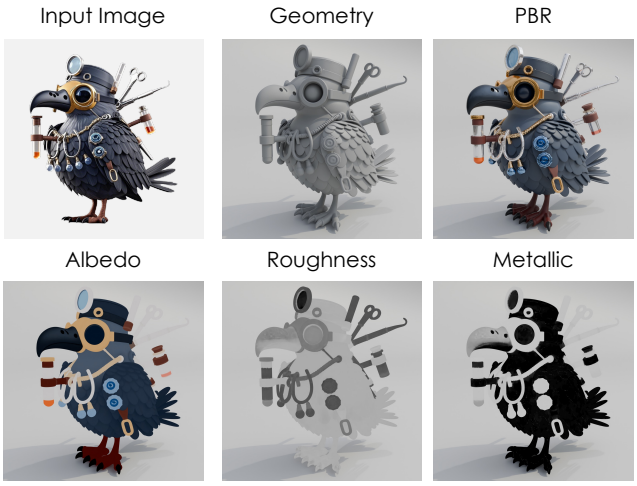

Figure 1. Illustration of our material generation results from a case study, with individual components visualized separately.

Figure 2. Illustration of our material generation results under different lightings, rendered using various environment maps.

⋆ Equal contribution. ‡ Corresponding authors.

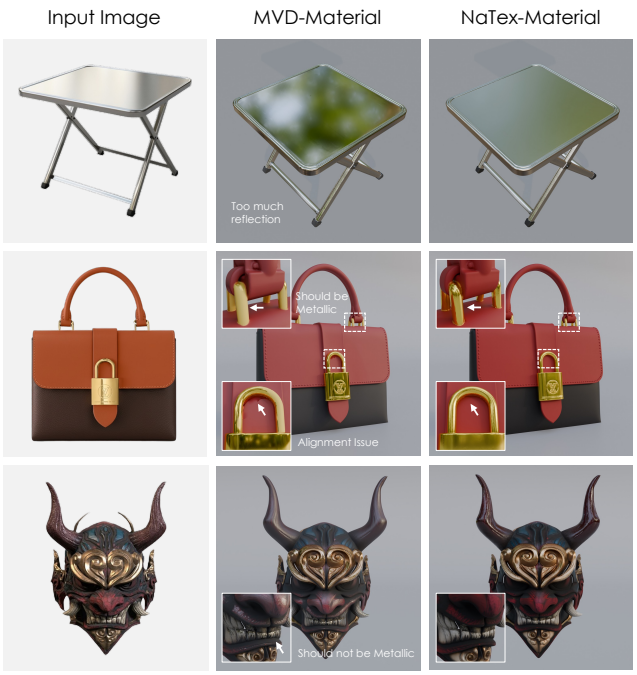

Figure 3. Visual comparison between our NaTex material generation pipeline and a conventional MVD-based material pipeline. Our method produces more accurate and better-aligned materials compared to prior approaches.

B. More Details on Applications

Material Generation. Thanks to the flexible design of the proposed NaTex framework, we can easily adapt it for material generation with color control. Specifically, we formulate material generation as a two-channel texture generation task conditioned on the textured mesh with albedo. We reuse the same color VAE employed for texture generation, representing roughness and metallic as two channels in an RGB color point cloud. A new material DiT is then trained on this material color point cloud data, conditioned on the input image (image control), the textured mesh with albedo (color control), and the input geometry (geometry control). During inference, we adopt a two-stage approach: the first stage predicts the albedo, and the second stage predicts roughness and metallic based on the previously predicted albedo.

The generation results of NaTex-Material inherit the advantages of native texture generation, producing well-aligned and coherent roughness and metallic maps, as shown in Fig.1. We believe this represents a significant advantage for developing next-generation material generation frameworks, since previous MVD approaches often struggle with alignment and sometimes misinterpret material properties, as illustrated in Fig.3.

Fig.2 presents our material generation results under different lighting conditions, demonstrating the effectiveness of the generated materials. Fig.4 showcases additional high-quality PBR-textured assets generated by NaTex, with albedo, roughness, and metallic maps all produced natively by our framework.

Part Segmentation. We find that our model can be readily applied to part segmentation by conditioning on a 2D mask, as indicated in the main paper. Specifically, this can be achieved by first performing semantic segmentation on the input RGB image using SAM[2]. We then directly apply our texture model, NaTex-2B, without any additional training, feeding in the 2D mask to obtain the textured mesh.

Nevertheless, this zero-shot strategy may produce fragmented or inconsistent results for complex structures. To address this, we finetune the base model on a dedicated dataset. Surprisingly, the results of the finetuned model are highly accurate even on complex cases, as shown in Fig.5, providing strong 3D segmentation with well-aligned boundaries. This further demonstrates the effectiveness and adaptation capability of our model.

Part Texturing. Texturing individual parts is just as straightforward as generating textures for the entire object. Unlike previous MVD approaches, which struggle with interior regions, our method naturally circumvents this issue by predicting color directly in 3D space for different part surfaces. Fig.6 shows part texturing results obtained by directly applying NaTex-2B. It can be observed that our model effectively handles occluded regions between parts and generates accurate textures for these areas. Fig.7 provides additional visual examples.

Texture Refinement. Our model can also serve as a second-stage refiner for MVD pipelines. This can be easily achieved by fine-tuning NaTex-2B with color control conditioned on an initial texture. In general, our refiner can correct various projection errors and automatically inpaint occluded regions, as illustrated in Fig. 8. Moreover, thanks to strong conditioning, this process can be performed in just five steps without any distillation, making it extremely fast and efficient for a wide range of downstream tasks.

C. Limitations and Future Works

It is exciting that the proposed NaTex advances texture generation, producing more seamless results and generalizing to a variety of applications. However, limitations remain that warrant further research. For example, the reconstruction quality of the VAE could be improved to support higher-resolution textures. Exploring more flexible multi-scale or adaptive patching schemes may be a promising direction for supporting higher-resolution texture modeling [4, 5]. Data curation should be enhanced for material generation. Part segmentation could be refined to reduce ambiguity and improve granularity. New methods are needed to handle closed surfaces in adjacent parts for part

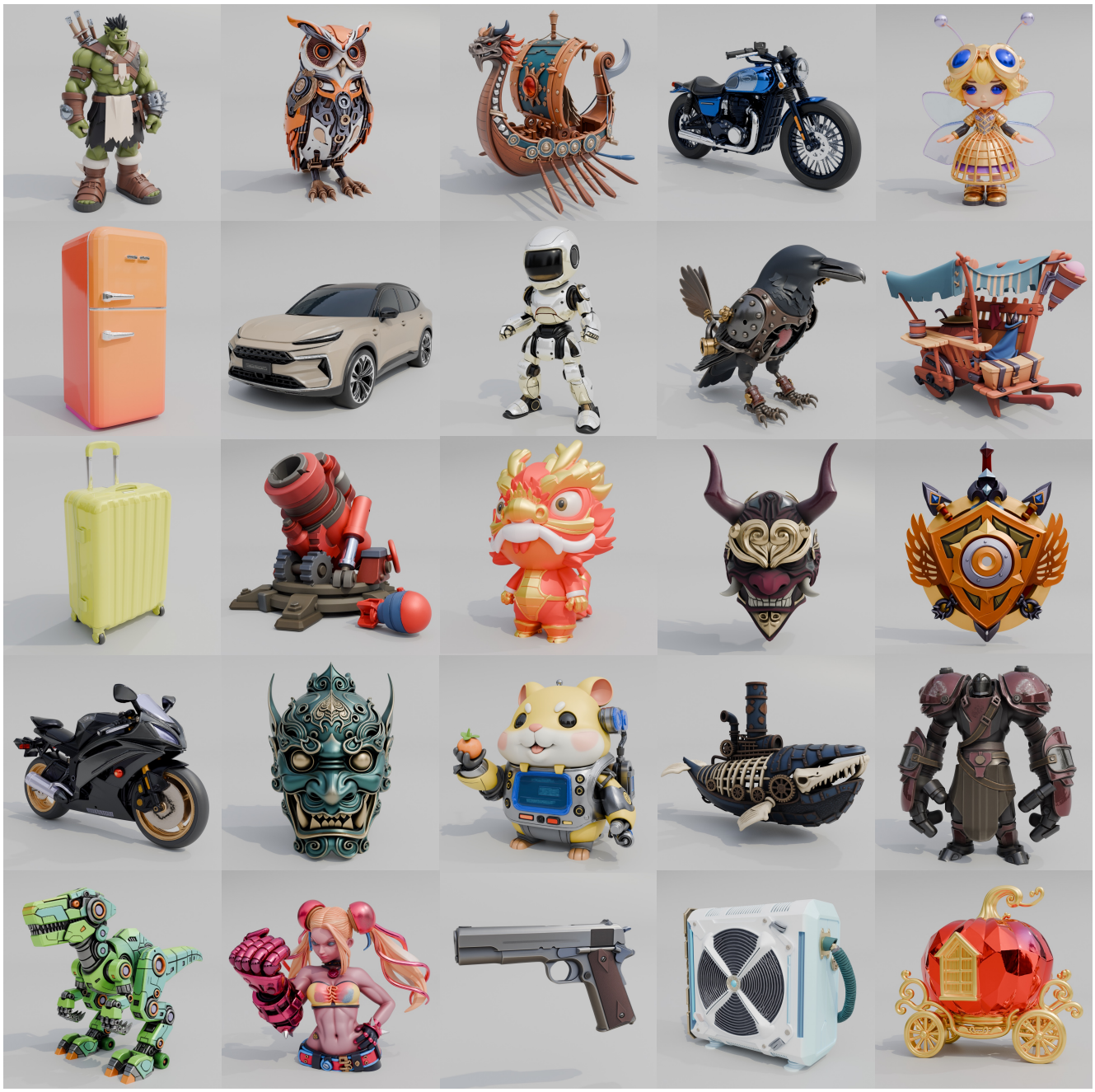

Figure 4. High-quality PBR-textured assets generated by NaTex. Geometry obtained from Hunyuan3D 2.5 [3].

texturing. Additionally, texture refinement also presents a promising direction for incorporating more 2D priors and leveraging established MVD research.

References

[1] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *arXiv preprint arXiv:2207.12598*, 2022. 1

[2] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 2

[3] Zeqiang Lai, Yunfei Zhao, Haolin Liu, Zibo Zhao, Qingxiang Lin, Huiwen Shi, Xianghui Yang, Mingxin Yang, Shuhui Yang, Yifei Feng, et al. Hunyuan3d 2.5: Towards high-fidelity 3d assets generation with ultimate details. *arXiv preprint arXiv:2506.16504*, 2025. 3

Figure 5. Visual results of part segmentation using a finetuned version of NaTex-2B. We provide a 2D mask as the input image for the given geometry, and NaTex textures the model accordingly.

[4] Wenzhuo Liu, Fei Zhu, Shijie Ma, and Cheng-Lin Liu. Mspe: multi-scale patch embedding prompts vision transformers to any resolution. *Advances in Neural Information Processing Systems*, 37:29191–29212, 2024. 2

[5] Wenzhuo Liu, Weijie Yin, Fei Zhu, Shijie Ma, Haiyang Guo, Xiao-Hui Li, Cheng-Lin Liu, et al. One patch doesn't fit all: Adaptive patching for native-resolution multimodal large language models. In *International Conference on Learning Representations*, 2026. 2

Figure 6. Illustration of part texturing using NaTex without any additional training. Our model generates textures for different parts without suffering from occlusion issues between them, as shown in the two renders with varying part arrangements.

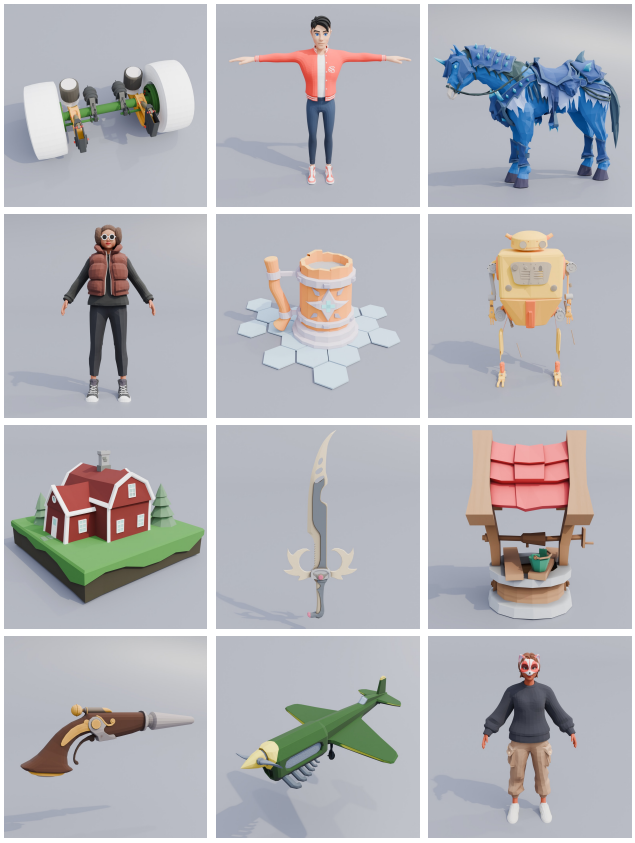

Figure 7. Visual examples of additional part texturing results generated by NaTex.

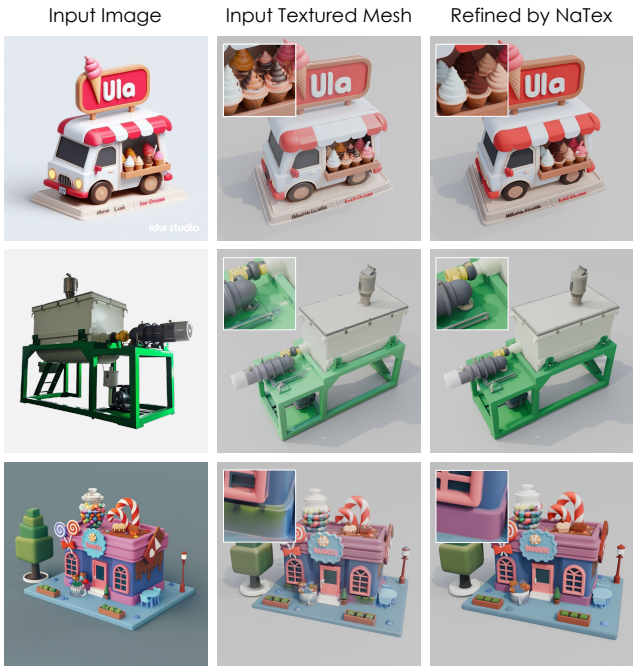

Figure 8. Illustration of texture refinement using NaTex with color control. As shown, NaTex effectively corrects errors in the input mesh caused by occluded regions and inconsistencies.