UniLat3D: Geometry-Appearance Unified Latents for Single-Stage 3D Generation

Supplementary Material

A.1. More Implementation Details

Uni-VAE. To accelerate and stabilize Uni-VAE training, we initialize $\mathcal{E}_{\text{sparse}}$ and $\mathcal{D}_{\text{sparse}}$ with the pretrained weights from TRELLIS. During the first 240k iterations, only $\mathcal{E}_{\text{dense}}$ and $\mathcal{D}_{\text{up}}$ are optimized, after which the entire Uni-VAE is trained end-to-end for an additional 90k iterations following TRELLIS. For the mesh decoder, we freeze $\mathcal{D}_{\text{uni}}$ and train our high-resolution mesh decoder from scratch. Unless otherwise specified, Adam [2] is used with a learning rate of 1×10^{-4}. The mesh decoder architecture is shown in Fig. 1.

UniLat Flow Transformer. For training the rectified flow models, we adopt DINOv3 [3] as the image encoder and apply classifier-free guidance [1] with a drop rate of 0.1. The model is first trained for 500k iterations with a batch size of 256 and a learning rate of 1×10^{-4}, and then fine-tuned for 160k iterations with a batch size of 1024 and a learning rate of 1×10^{-5}.

A.2. Model Architecture

In this section, we mainly provide the model architecture about our Uni-VAE $\{\mathcal{E}_{uni}, \mathcal{D}_{uni}\}$ and UniLat generation model $\mathcal{F}$.

A.2.1. Uni-VAE

For the sparse encoder $\mathcal{M}_{sparse}$, we mainly follow TRELLIS's configurations to build a sparse Transformer. For the dense encoder $\mathcal{M}_{dense}$, a set of conv3D layers is used as the main architecture. The settings of $\mathcal{E}_{sparse}$, $\mathcal{D}_{up}$ are shown in Table 1 and details of $\mathcal{E}_{uni}$ are provided in Table 2.

Table 1. Model details of Uni-VAE modules $\mathcal{M}_{dense}, \mathcal{D}_{up}$. "Channels" denotes model channels after each up/downsampled convolution layer.

Model	ResBlocks	Channels
$\mathcal{E}_{sparse}$	4	$[32, 128, 512]$
$\mathcal{D}_{up}$	4	$[512, 128, 32]$

Table 2. Model details of Uni-VAE modules $\mathcal{M}_{sparse}, \mathcal{D}_{gs,mesh}$.

Model	Latent Res.	Model Channels	Latent. Channels	Blocks	Attn. Heads	Window Size
$\mathcal{M}_{sparse}, \mathcal{D}_{sparse}$	64	768	8	12	12	8

A.2.2. UniLat Flow Transformer

Structure details about our UniLat flow Transformer $\mathcal{F}_{uni}$ are provided in the Table 7. The main architecture of $\mathcal{F}_{uni}$ is similar to TRELLIS's sparse structure flow Transformer. The input noise ϵ would be flattened to 1D tensors. Positional encoding is applied to a flattened tensor, and it would be fed to Transformer blocks with self&cross-attention layer and modulated by condition signal & timestamps. Finally, the flattened tensor would be unpatchified to 3D results, the shape is the same as ϵ.

Table 3. Ablation study on the visual encoder for condition images.

Model	Cond. Encoder	CLIP↑	$\text{FD}_{\text{dinov2}}$↓
UniLat3D	DINOV2	**90.83**	52.58
UniLat3D	DINOV3	90.60	**49.90**

A.3. More Ablation Studies

Resolution of Latents & Final Representation. We explore the latent space of reconstruction quality in Uni-VAE. We train Uni-VAE at different latent resolutions, including 8^3, 16^3, and 32^3. As shown in Table 4, higher UniLat resolutions lead to better reconstruction results. Note that our Uni-VAE achieves similar or even better reconstruction performance than TRELLIS with smaller resolutions. Noted that our UniLat also encodes the empty space. We found that in such low latent resolution at 16^3, UniLat demonstrates superior reconstruction quality compared with TRELLIS. Meanwhile, the generative model in 16^3 would be better than the model trained in 32^3; it may be because the reconstruction-generation trade-off [4] happens in such a resolution. We also provide ablation studies of the high-resolution mesh decoder in Table 4, demonstrating that upsampling structure latents in resolution 512 would enjoy better performance.

Visual Encoder of Condition Images. Recently, DINOv3 [3] emerges as a strong visual encoder model that could extract high-quality details from the image. We compare the performance between DINOv2 and DINOv3 for encoding condition images. Flow models with different visual encoders are trained for 500k iterations and tested on Toys4K. In our experiments, the flow Transformer with the DINOv3 encoder shows better quality on complex object

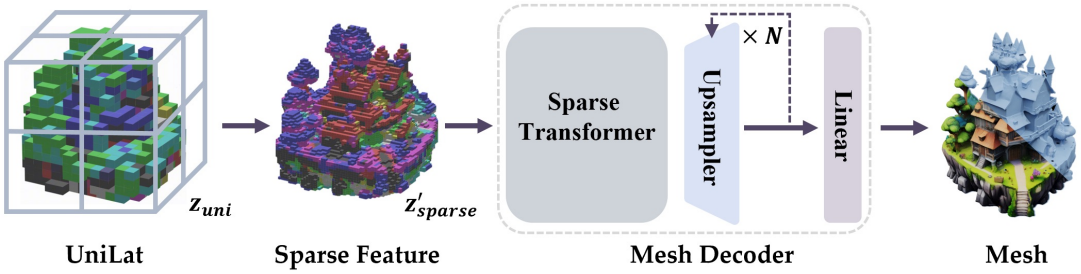

Figure 1. Mesh decoder architecture.

Table 4. VAE reconstruction results with latents of different resolutions. "LR": UniLat resolution. "RR": Output representation resolution.

Model	LR.	RR.	PSNR↑	SSIM↑	LPIPS↓
TRELLIS (Mesh)	64^3	256	31.91	97.44	0.0328
UniLat3D (Mesh)	16^3	256	32.01	97.56	0.0319
UniLat3D (Mesh)	16^3	512	32.35	98.03	0.0305
TRELLIS (GS)	64^3	64	34.74	98.52	0.0146
UniLat3D (GS)	8^3	64	33.51	98.13	0.0200
UniLat3D (GS)	16^3	64	34.80	98.49	0.0158
UniLat3D (GS)	32^3	64	**34.92**	**98.53**	**0.0145**

Table 5. Efficiency comparison on H800 GPU with FlashAttention-3. FID: FD_{DINOv2} on Toys4K.

Model	Steps	Time (s)	FID↓
TRELLIS	25+25	3.1	52.54
UniLat3D	25	**2.1**	**49.02**

Table 6. Resource utilization across different latent resolutions.

Model	Steps	Time	VRAM
TRELLIS	25+25	3.10s	6.4GB
UniLat3D-16^3	35	3.05s	5.9GB
UniLat3D-32^3	35	90s	10.5GB
UniLat3D-64^3	35	1h	16GB

generation, which leads to a better FD_{dinov2} result as shown in Table 3.

Efficiency Analysis. To demonstrate that our dense design is highly amenable to modern hardware acceleration, we conduct efficiency experiments on an H800 GPU with FlashAttention-3. As shown in Table 5, UniLat3D with 25 denoising steps achieves an FID of 49.02 in 2.1 seconds, surpassing TRELLIS (FID 52.54 in 3.1 seconds with 25+25 steps for two-stage generation). This demonstrates that our unified dense architecture benefits significantly from hardware acceleration and achieves better quality-efficiency trade-offs.

Resource Utilization Analysis. We analyze the computational costs of UniLat3D at different latent resolutions as shown in Table 6. Our 16^3 configuration achieves comparable inference time (3.05s) and lower VRAM usage (5.9GB) compared to TRELLIS (3.10s, 6.4GB). Scaling to 32^3 (90s, 10.5GB) or 64^3 (1h, 16GB) offers limited reconstruction gains but incurs prohibitive costs. This demonstrates that our 16^3 latent resolution achieves an optimal balance between quality and efficiency, benefiting from the high compression capability of latent diffusion models.

A.4. More Visualizations

We also provide more visualization with other commercial models in Fig. 2 and Fig. 4.

Table 7. Model details of UniLat3D flow Transformer.

Model	Params	Latent Res.	Latent Channels	Model Channels	Cond. Channels	Blocks	Attn. Heads
$\mathcal{F}_{uni}$	1.30B	16	32	1280	1280	36	32

References

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

[2] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[3] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025. 1

[4] Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15703–15712, 2025. 1

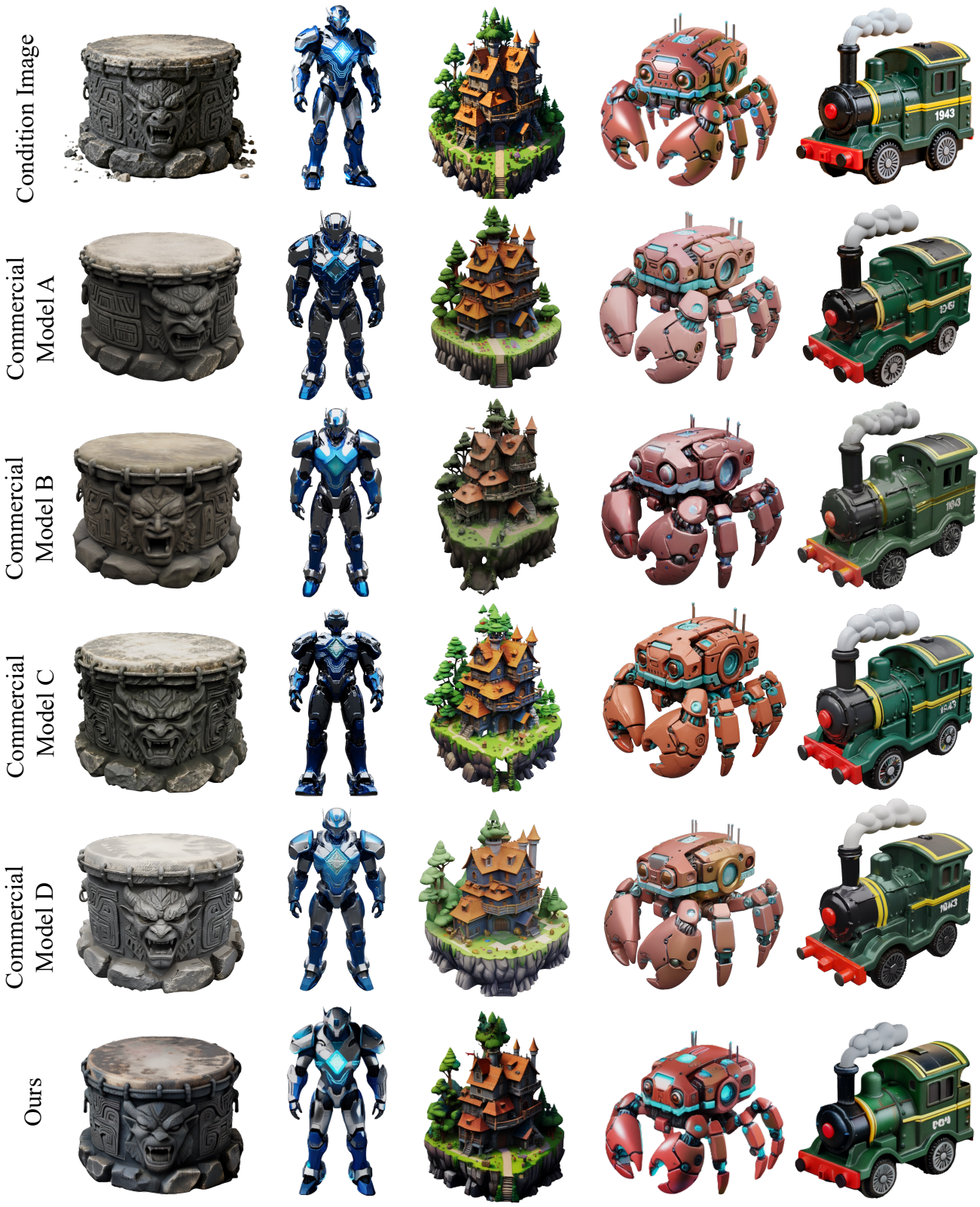

Figure 2. Qualitative comparisons with commercial models. Our UniLat3D shows competitive performance even with only publicly available training data.

Figure 3. 3D mesh assets generated by our UniLat3D.

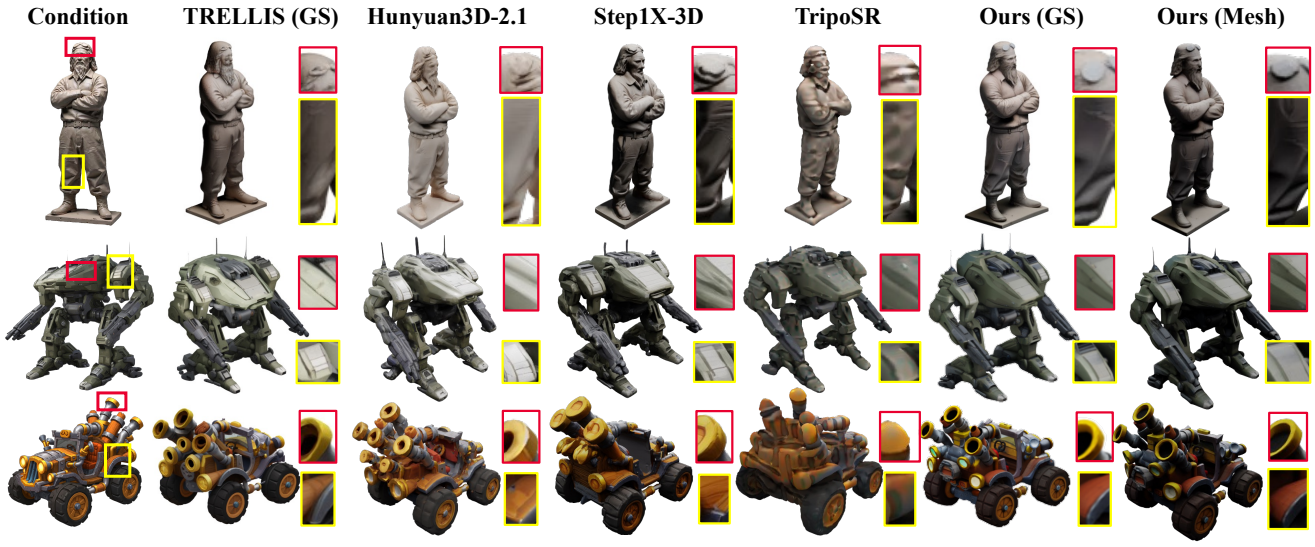

Figure 4. Qualitative comparisons with other methods. Thanks to our unified representation, UniLat3D achieves superior performance and better correspondence with input images.