AssetFormer: Modular 3D Assets Generation with Autoregressive Transformer

Lingting Zhu[1] Shengju Qian[2] Haidi Fan[2] Jiayu Dong[2] Zhenchao Jin[1]

Siwei Zhou[2] Gen Dong[2] Xin Wang[2] Lequan Yu[1]

[1]The University of Hong Kong [2]LIGHTSPEED

Introduction

The Problem:

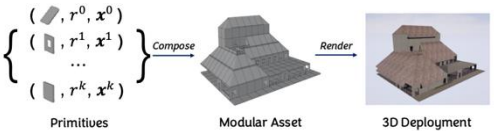
- Existing 3D representations struggle to meet the high-quality standard by modern games
- UGC and online gaming requires efficient representations

The Need:

- Games & UGC require modular, compact, engine-ready assets
- Efficient storage & transmission
- Accessible creation for non-professionals

Our Solution: AssetFormer

- Autoregressive Transformer for modular 3D asset generation from text



Modular assets: primitives with class c, rotation r, position x

Key Insight: Assets as Sequences of Primitives

Core Idea:

- 3D assets = a sequence of modular primitives
- Each primitive P_j parameterized by:

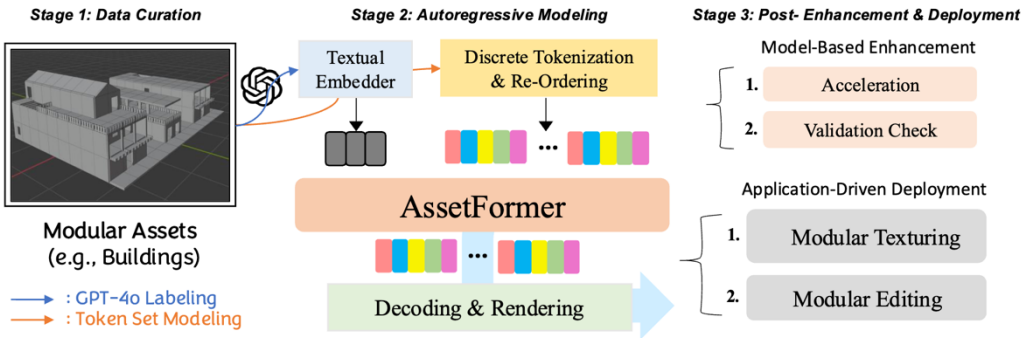
$P = (c, r, x)$, where c: class, r: rotation, x: position

Why Sequences?

- Discrete attributes → lossless tokenization
- Natural fit for autoregressive Transformers
- Mirrors step-by-step construction

Goal: Learn $G : t \rightarrow \{P\}$ from text t

AssetFormer: Framework Overview



Pipeline: Render assets → GPT-4o captioning → Text-conditioned AR training → Engine-ready assets

Discrete Tokenization & Token Re-Ordering

Joint Vocabulary:

$$\mathcal{V} = \mathcal{C} \vee \mathcal{R} \vee \mathcal{X}_0 \vee \mathcal{X}_1 \vee \mathcal{X}_2 \vee \{< \text{EOS} >\},$$
$$|\mathcal{V}| = |\mathcal{C}| + |\mathcal{R}| + |\mathcal{X}_0| + |\mathcal{X}_1| + |\mathcal{X}_2| + 1,$$

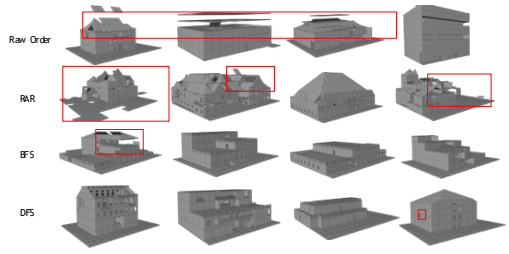
Token Sequence:

$$T = \{c^0, r^0, x_0^0, x_1^0, x_2^0, \ldots, c^{n-1}, r^{n-1}, x_0^{n-1}, x_1^{n-1}, x_2^{n-1}, \text{EOS}\},$$

Token Re-Ordering (Key Design):

- Graph traversal: DFS/BFS on connectivity
- Captures hierarchical & spatial structure
- DFS empirically performs best

$$T' = \text{ReOrder}(T) = \{c^{\tau_0}, r^{\tau_0}, x_0^{\tau_0}, x_1^{\tau_0}, x_2^{\tau_0}, \ldots, c^{\tau_{n-1}}, r^{\tau_{n-1}}, x_0^{\tau_{n-1}}, x_1^{\tau_{n-1}}, x_2^{\tau_{n-1}}, \text{EOS}\}.$$



Token ordering impact: DFS preserves local structure and modular connectivity

Training Objective & Decoding Strategies

Training:

- Decoder-only Transformer (Llama-based)
- Next-token prediction with cross-entropy

$$\mathcal{L} = \text{CrossEntropy}(\text{Shift}(\hat{S}), \text{Tokenize}(\{P\})),$$

- Text features via FLAN-T5 XL
- Rotary Positional Embeddings (RoPE)
- Classifier-Free Guidance (CFG)

Inference:

- Top-k (k=10, temp=0.7, CFG=2.0)
- Vocabulary-aware logit filtering

SlowFast Decoding:

- Adapted from speculative decoding
- **Draft** (87M): fast, easy tokens
- **Target** (312M): complex tokens
- 1.48× speedup, no quality loss

Model Configurations	FID ↓	Speed (token/s) ↑
AssetFormer-S (87M)	60.420	151.31
AssetFormer-B (312M)	55.186	80.62
SlowFast Decoding	55.831	119.02

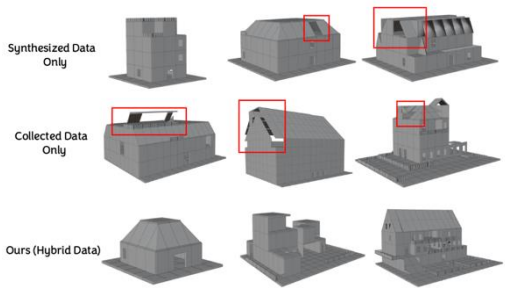
Dataset: Real-World Modular 3D Assets

Data Sources:

- 16K real user-created assets (UGC platform)
- 4K procedurally synthesized (PCG)
- Complementary: PCG → structure; real → diversity

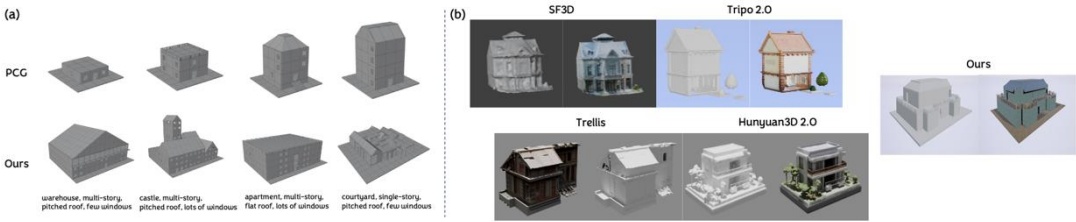
Data Processing:

- Mapped to 25 primitives (roof, wall, etc.)
- Avg. token length >4,000; up to 1K primitives
- GPT-4o captioning for text conditions
- Automatic + manual quality filtering



Combining both data sources yields best quality and diversity

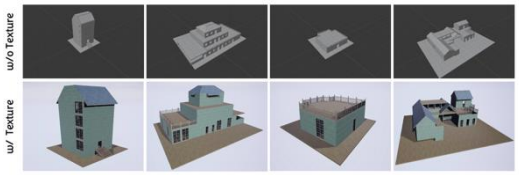
Qualitative Results: Comparison with Baselines



(a) **PCG:** limited diversity, no text control (b) **3D Gen:** dense meshes, imperfect surfaces

AssetFormer: modular, text-controlled, precise

Texture Mapping & Engine Integration



Textured buildings at various levels of detail

Modular Advantage:

- Primitives natively support texturing
- Easy post-processing & user friendly
- Procedural & generative texturing

Engine-Ready:

- Seamless Unreal Engine integration
- Compact for online transmission

Gallery rendered in Unreal Engine

Quantitative Evaluation & Ablation Studies

Methods	FID ↓	CLIP ↑
True Data	/	**0.322**
PCG (Algorithm 1)	108.476	0.319
AssetFormer + Greedy Search	63.351	0.319
AssetFormer + Beam Search	63.333	**0.321**
AssetFormer + Top-K Sampling	**55.186**	0.320

Training Data Types	FID ↓	CLIP ↑
Synthesized Data Only	113.560	0.320
Collected Data Only	63.381	**0.321**
Synthesized Data + Collected Data	**55.186**	0.320

Ordering Techniques	FID ↓	CLIP ↑
Raw Order	65.215	0.318
RAR (Yu et al., 2024)	83.561	0.313
Breadth-First-Search	61.620	0.319
Depth-First-Search	**55.186**	**0.320**

Method	Compactness	Diversity	Aesthetic	Complexity
Ground Truth	*3.83*	*4.00*	*3.67*	*4.42*
PCG	**4.47**	2.42	3.33	2.08
AssetFormer	3.42	**3.50**	**3.50**	**3.92**

Conclusion & Future Work

Summary:

- First AR Transformer for modular 3D assets
- Assets as discrete primitive sequences
- DFS token re-ordering for spatial structure
- SlowFast decoding
- Large-scale real-world modular 3D dataset

Impact:

- Bridges generative AI & industrial 3D
- Engine-ready assets for games & UGC
- Zero-shot editing & diverse generation

Future Directions:

- Image-based conditioning
- Scaling to complex scenes

Thank You!

Code:
https://github.com/Advocate99/AssetFormer

Contact: ltzhu99@connect.hku.hk