

HASAI 3D: Deterministic vs Probabilistic Methods

Dragoslav “Drago” Pavlovic

6th-january-2026

Author: Dragoslav “Drago” Pavlovic @drago3dx <https://drago3d.com/>

HASAI3D Deterministic vs Probabilistic Methods

This whitepaper explores the differences between different methods used in the context of applying Artificial Intelligence for generating production-ready 3D models. It provides insights into the advantages and limitations of each approach for different applications. Also, it covers the possibility of combining both approaches into a hybrid method. The hybrid approach is explained through an example of HASAI 3D.

Table of Contents

1. Overview
2. Probabilistic Method
3. Deterministic Method
4. Hybrid Method
5. HASAI 3D – Hybrid Approach Smart Artificial Intelligence
6. Method Comparison Table
7. Applications
8. Business Edge
9. Vision
10. Conclusion
11. Technical Appendix
12. Glossary

Overview

3D graphics is a complex medium and arguably one of the most expensive ones to produce. As with any expensive process, there is a constant drive to make it **more efficient, more scalable, and less costly**, while retaining acceptable quality. This applies across many industries: video games, film, advertising, architecture, construction engineering, simulation, and even medicine. Although 3D graphics is related to other computer graphics mediums, such as 2D images and video, the similarity is often overstated. A video is ultimately a sequence of 2D frames. A 3D model, however, is a **structured spatial representation** that must satisfy far stricter constraints. Generative AI has seen rapid adoption in 2D image and video generation. The success of these tools has led many to assume that the same underlying techniques can be directly applied to 3D. This document does not focus on subjective opinions about generative AI. Instead, it focuses on **objective technical constraints** specific to 3D graphics. The dominant technology behind modern generative AI is machine learning trained on large datasets. These systems generate outputs based on **probability**

distributions, not fixed rules. They do not guarantee a specific result — they sample from a space of likely outcomes. This probabilistic nature is acceptable, and often desirable, in 2D content generation. In 3D graphics, however, this same property becomes a fundamental limitation.

Probabilistic Method

A probabilistic method generates outputs by estimating the likelihood of certain results based on training data.

In the context of 3D, this typically means:

- Inferring geometry from images or point clouds
- Approximating shapes based on learned patterns
- Producing outputs that are visually plausible, but not structurally guaranteed

Common characteristics of probabilistic 3D generation include:

- Non-deterministic outputs (same input \neq same result)
- Dense, noisy, or irregular geometry
- Missing or unusable UVs
- No semantic understanding of parts or hierarchy
- Limited or no editability

From a production standpoint, these outputs are rarely usable without extensive manual cleanup. They may look correct at first glance, but they lack the structural consistency required for real-time rendering, animation, physics simulation, or manufacturing workflows.

Probabilistic methods optimize for visual appearance, not correctness.

Deterministic Method

A deterministic method produces the same output every time for a given input.

In traditional 3D pipelines, determinism is the norm:

- Procedural modeling
- Parametric design
- CAD systems
- Geometry scripting
- Rule-based asset generation

Deterministic systems operate on explicit rules and constraints:

- Geometry is defined, not guessed
- Topology is intentional
- UVs, materials, pivots, and scale are known in advance
- Outputs are predictable and reproducible

The downside is that purely deterministic systems are often:

- Slow to author
- Rigid
- Dependent on human expertise
- Difficult to scale without significant upfront effort

While deterministic methods produce fully production-ready assets, they traditionally lack the flexibility and automation promised by AI-driven approaches.

Hybrid Method

The hybrid method combines **probabilistic intelligence** with **deterministic execution**.

Instead of using AI to directly generate final geometry, AI is used to:

- Interpret intent
- Reason about structure
- Select rules and parameters
- Drive deterministic systems

The final 3D output is not sampled from noise, but **constructed through rules**.

This approach preserves the strengths of both methods:

- Flexibility and adaptability from AI
- Precision, reproducibility, and correctness from deterministic systems

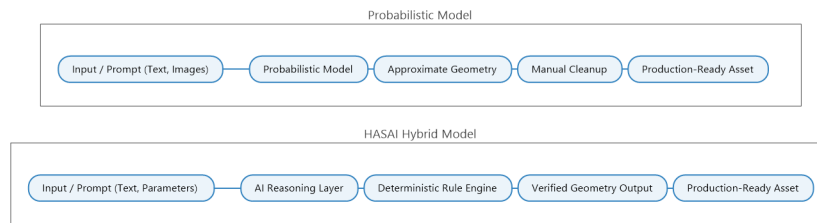


Figure 1: Probabilistic vs HASAI Hybrid Model

Figure 1: Comparison between a probabilistic 3D generation pipeline and the HASAI hybrid deterministic pipeline.

HASAI 3D – Hybrid Approach Smart Artificial Intelligence

HASAI 3D is a hybrid AI system designed specifically for **production-ready 3D generation**.

In HASAI 3D:

- AI does not generate meshes directly
- AI reasons about *what* needs to be built
- Deterministic systems handle *how* it is built

The result is:

- Identical output for identical input
- Clean, editable geometry
- Predictable cost and performance
- Native compatibility with real-time engines and DCC tools

HASAI 3D treats 3D not as an image reconstruction problem, but as a **logical construction problem**.

Method Comparison Table

Method	Probabilistic Method	Deterministic Method	Hybrid Method
Production readiness	Not production ready	Fully production ready	Fully production ready
Final cost per asset	~\$1 + cleanup	\$50–5,000	~\$0.002 (no training)
Generation speed	Minutes	Days	Seconds
Editability	None	Full	Full
Consistency	Non-deterministic	Deterministic	Deterministic
Scalability	Limited	High	High

Applications

The distinction between probabilistic and deterministic approaches becomes most apparent in real-world applications where **consistency, editability, and scale** are required.

Deterministic or hybrid approaches like HASAI 3D are particularly suited for:

- **Real-time applications**

Game engines and interactive environments require predictable geometry, stable topology, and strict performance constraints.

- **Mass asset generation**

Furniture, props, architectural elements, and modular environments where thousands of variants must follow the same structural rules.

- **Configurators and digital twins**

Systems where geometry must directly reflect parameters such as dimensions, materials, pricing, or physical constraints.

- **Simulation and engineering workflows**

Use cases where approximation is unacceptable and geometry must be logically correct.

Probabilistic methods may still be useful for **concept exploration** or visual prototyping, but they do not replace deterministic pipelines where assets must be deployed at scale.

Business Edge

From a business perspective, determinism is not just a technical preference — it is an **economic advantage**.

Deterministic systems offer:

- **Predictable cost**

Asset generation cost does not scale with randomness, retries, or cleanup labor.

- **Lower infrastructure requirements**

Logic-driven systems reduce dependency on large GPU clusters and repeated inference cycles.

- **Pipeline compatibility**

Outputs integrate directly into existing DCC tools and real-time engines without manual intervention.

- **Scalability**

Once rules are defined, generating one asset or one million assets follows the same process.

HASAI 3D is designed around these constraints, enabling 3D asset generation to behave more like **manufacturing** than experimentation.

Vision

The long-term goal of HASAI 3D is not to replace artists or engineers, but to **formalize 3D knowledge into systems**.

As 3D content becomes increasingly central to digital products, the ability to generate assets that are:

- deterministic,
- explainable,
- and structurally correct

will become more important than purely visual novelty.

HASAI 3D positions itself as a foundation for this transition — treating 3D as a **logical medium**, not a probabilistic guess.

Conclusion

Probabilistic methods have demonstrated impressive results in 2D content generation. However, their limitations become evident when applied to 3D graphics, where correctness, structure, and consistency matter more than visual plausibility.

Deterministic systems remain the only reliable way to produce production-ready 3D assets, but they traditionally lack flexibility and automation.

Hybrid systems like HASAI 3D combine the strengths of both approaches, enabling scalable, cost-efficient, and deterministic 3D generation suitable for real-time and industrial pipelines.

Technical Appendix

Determinism and Reproducibility

In a deterministic system, identical inputs produce identical outputs.

This property enables version control, debugging, and large-scale automation — all of which are critical in production pipelines.

HASAI 3D enforces determinism at the geometry construction level, ensuring reproducible results across environments.

Topology Guarantees

Production-ready 3D assets require predictable topology for animation, physics, and rendering.

Rather than inferring topology from data, HASAI 3D generates topology through explicit construction rules, ensuring consistency and editability.

Semantic Hierarchy

HASAI 3D models maintain awareness of object structure:

- parts
- pivots
- materials
- scale relationships

This enables downstream operations such as configuration, replacement, and parameter-driven variation.

Validation and Constraints

Deterministic generation allows outputs to be validated against predefined constraints, such as:

- polygon budgets
- scale limits
- naming conventions
- engine-specific requirements

This validation step is not possible in purely probabilistic systems.

Glossary

3D model – A structured digital representation of an object in three-dimensional space, typically consisting of geometry, materials, UVs, and metadata required for rendering, simulation, or manufacturing.

Probabilistic method – A system that generates outputs by sampling from probability distributions learned from data, without guaranteeing identical results for identical inputs.

Deterministic method – A system that produces identical outputs for identical inputs, based on explicit rules and constraints.

Production-ready asset – A 3D asset that meets the technical requirements of real-time engines, simulation systems, or manufacturing pipelines without requiring manual cleanup.