LoST: Level of Semantics Tokenization for 3D Shapes

Supplementary Material

1. Additional Qualitative Results

We provide an extensive gallery of qualitative results in the accompanying ***supplemental webpage***, illustrating the capabilities of our tokenizer and autoregressive (AR) model. These 3D visualizations demonstrate that our method produces high-fidelity reconstructions that visually surpass recent baselines. Note our AR model's flexibility in generating complete and plausible 3D shapes even when conditioned on a few tokens. These qualitative findings are consistent with the strong quantitative performance reported in Table 2 of the main manuscript, further validating the effectiveness of our semantic tokenization strategy.

2. Shape Retrieval using RIDA

To quantitatively validate that our Relational Inter-Distance Alignment (RIDA) objective successfully reorganizes the 3D latent space according to semantic salience rather than just geometric proximity, we evaluate our method on a shape retrieval task. Since RIDA is designed to distill the semantic topology of the DINOv2 [4] (*teacher*) space into 3D triplanes, we utilize DINO similarity as the ground truth for defining semantic neighbors.

We compare our RIDA-aligned features against a baseline of (i) raw triplane latents, which primarily capture geometric spatial structure and (ii) a Direct Regression baseline, trained to predict DINO features via explicit supervision. To assess generalization, we conduct this evaluation on two distinct datasets: (i) our In-Distribution set, consisting of held-out samples from our training distribution, and (ii) our Evaluation Set (Out-of-Distribution), which contains shapes generated by the unseen Step1X-3D model with a different underlying representation as specified in Section 4.1 in the manuscript. For the Direct Regression baseline, we employ the same transformer backbone as RIDA but replace the relational contrastive and distillation objectives with a direct spatial regression loss (MSE). Empirically, we observe that this direct mapping is ineffective; the network suffers from optimization stagnation, exhibiting early loss plateauing on the validation set and failing to capture discriminative semantic features.

We report Recall@K to measure the proportion of ground-truth semantic neighbors successfully retrieved, and Mean Average Precision (mAP@K) to evaluate the quality of their ranking within the top results. Additionally, we compute the Jaccard Index to quantify the set intersection-over-union (IoU) between the retrieved candidates and the ground truth.

As shown in Table 1, RIDA demonstrates superior semantic alignment compared to the geometric baseline. On the challenging OOD evaluation set , our method significantly improves Mean Average Precision (mAP@3) from 17.47% to 28.28%, proving that RIDA captures abstract semantic identity that is robust to low-level geometric variations. This performance gap is further amplified on the in-distribution set (benefiting from the same VAE latent encoding), where RIDA achieves a 44.28% mAP compared to the baseline's 16.42% (triplane) and 22.90% (feature regression). These results confirm that while raw triplanes are limited to geometric matching, RIDA effectively aligns 3D shapes with the rich semantic hierarchy of DINO. The results on the in-distribution set are critical for training LoST. RIDA also significantly surpasses our direct regression baseline that is trained to predict DINOv2 features.

Table 1. **Shape Retrieval Evaluation.** We evaluate RIDA against raw triplane latents and a direct regression baseline that is trained to predict DINOv2 features. Ground truth neighbors are defined by DINOv2 similarity. RIDA (ours) outperforms the geometric baseline on both the in-distribution validation set and the out-of-distribution evaluation set (generated using Step1X-3D), confirming that RIDA effectively captures semantics.

	Out-of-Distribution Set			In-Distribution Set		
Metric	**Triplane**	**Regression**	**RIDA (ours)**	**Triplane**	**Regression**	**RIDA (ours)**
Top-3 Retrieval (K=3)						
Recall@3	20.20%	20.63%	**32.03%**	19.07%	27.00%	**48.50%**
mAP@3	17.47%	17.28%	**28.28%**	16.42%	22.90%	**44.28%**
Jaccard	13.73%	13.91%	**23.25%**	13.60%	19.19%	**38.36%**
Top-5 Retrieval (K=5)						
Recall@5	19.54%	20.70%	**30.30%**	18.48%	26.80%	**47.90%**
mAP@5	15.12%	15.29%	**24.71%**	14.42%	21.16%	**41.85%**
Jaccard	12.42%	13.22%	**20.49%**	12.01%	17.80%	**35.77%**

3. Ablation on RIDA

Following recent advances in image tokenization [1, 10], which leverage alignment losses like REPA [13] to structure latent spaces, we integrate RIDA to explicitly align our 3D triplane representations with semantic priors. While the diffusion decoder possesses an inherent capacity to move towards plausible geometry via its generative prior—especially at lower guidance scales—we find that RIDA significantly augments this capability. By enforcing a structured relationship between the 3D latent space and the teacher's semantic embedding, RIDA serves as a potent regularizer that enhances the semantic consistency of the decoded shapes. Furthermore, the semantic supervision acts as a stabilizing factor, counteracting the inherent training volatility introduced by the nested dropout mechanism.

This benefit is most pronounced in low-bitrate regimes. As demonstrated in Table 2 particularly DINOv2 similarity scores and FID, when the token budget is constrained,

Table 2. **Ablation tokenizer reconstruction.** We compare LoST trained without the proposed RIDA semantic alignment across multiple decoding levels. RIDA consistently improves semantic reconstruction quality, as measured by DINO and DINOv2 similarity, with the largest gains appearing in the low-token regime where semantic guidance is most critical. This confirms that RIDA helps short token prefixes encode more semantically meaningful structure. The best score at each decoding level is highlighted in **bold**.

	w/o RIDA					**w/ RIDA (ours)**				
Num Tokens →	**1**	**4**	**16**	**64**	**512**	**1**	**4**	**16**	**64**	**512**
DINO↑	0.720	0.758	**0.821**	0.876	0.904	**0.731**	**0.765**	0.814	**0.880**	**0.921**
DINOv2↑	0.528	0.590	0.693	0.763	0.867	**0.556**	**0.612**	**0.694**	**0.805**	**0.875**

the model cannot rely solely on dense geometric encoding. In these settings, RIDA effectively bridges the gap between high-level semantic intent and geometric reconstruction, yielding substantial quantitative gains and ensuring that even short token prefixes decode into semantically recognizable structures. The Chamfer Distance remains similar in both settings, which suggests that utilizing RIDA does not negatively impact training for geometry but enhances semantic alignment. We further note that extended training of the diffusion decoder eventually leads to convergence without RIDA, our method accelerates this process (∼40% faster). These findings are consistent with Flextok's ablation study [1] when using REPA.

Note our ablation of a direct regression baseline that is trained to predict DINO features directly in Table 1; this approach fails to accurately learn semantics. We present qualitative results on the ablation in the **supplemental webpage**.

4. Details about RIDA

Inter-Instance Rank Distillation. The contrastive loss enforces separation based on hard thresholds between positive and negative samples, but discards the rich, continuous relational structure within the teacher's space. This continuous structure is essential, but it is non-trivial to transfer to the student space. To this end, we are inspired by Relational Knowledge Distillation (RKD) [5], which transfers pairwise Euclidean distances. In our setting, we use cosine similarities $\mathbf{c}_i^s := [c_{ij}]_{z_j \in \mathcal{B} \text{ with } i \neq j}$ and the corresponding similarities in the teacher space $\mathbf{c}_i^t$. However, in our cross-modal setting (3D-to-2D), absolute similarities are not directly comparable; a naive loss on raw cosine similarities ($\|\mathbf{c}_i^s - \mathbf{c}_i^t\|_2^2$), fails to converge, as the student and teacher's per-anchor similarity distributions (i.e., their means $\mu(\cdot)$ and scales $\sigma(\cdot)$) are fundamentally misaligned. We therefore introduce a rank distillation loss, which is designed to be *invariant* to these modality-specific affine transformations. Instead of matching individual pairs, we match the *entire* per-anchor similarity vector. We achieve invariance by standardizing (z-scoring) each anchor's similarity row independently to remove its specific mean and scale, thus isolating the pure relational pattern:

$$\widetilde{\mathbf{c}}_i^s = \frac{\mathbf{c}_i^s - \mu(\mathbf{c}_i^s)}{\sigma(\mathbf{c}_i^s)}, \qquad \widetilde{\mathbf{c}}_i^t = \frac{\mathbf{c}_i^t - \mu(\mathbf{c}_i^t)}{\sigma(\mathbf{c}_i^t)}. \tag{1}$$

The loss is the Mean Squared Error between these z-scored, distribution-invariant vectors:

$$\mathcal{L}_{\text{rank}} := \mathbb{E}_{\mathbf{z}_i \in \mathcal{B}} \left[\left\| \widetilde{\mathbf{c}}_i^s - \widetilde{\mathbf{c}}_i^t \right\|_2^2 \right]. \tag{2}$$

This objective is mathematically proportional to maximizing the Pearson correlation coefficient for each row, as $\|\widetilde{a} - \widetilde{b}\|_2^2 \propto (1 - \text{corr}(a, b))$ for z-scored vectors $\widetilde{a}$ and $\widetilde{b}$. By factoring out the per-anchor mean and standard deviation, $\mathcal{L}_{\text{rank}}$ purely optimizes for the *relative neighborhood ranking*, which is the core semantic relation we distill.

Spatial Structure Distillation. To ensure the student's spatial tokens $\mathbf{S}_i^s$ capture the same part-level relationships as the teacher's $\mathbf{S}_i^t$, we distill the intra-instance token affinities. Instead of a direct L2 match, we match the *distribution* of similarities. We compute self-affinity matrices $\mathbf{A}_i^s, \mathbf{A}_i^t \in \mathbb{R}^{K \times K}$ within each instance i by:

$$\mathbf{A}_i[k, \ell] = \langle \mathbf{S}_{i,k}, \mathbf{S}_{i,\ell} \rangle. \tag{3}$$

We then apply a row-wise softmax to create affinity distributions, $\mathbf{a}_{i,k} = \text{Softmax}(\mathbf{A}_i[k, \cdot])$. The spatial loss minimizes the KL divergence between the teacher and student distributions for each token:

$$\mathcal{L}_{\text{spatial}} := \mathbb{E}_{i,k} \left[D_{\text{KL}} \left(\mathbf{a}_{i,k}^t \,\|\, \mathbf{a}_{i,k}^s \right) \right]. \tag{4}$$

This forces the student's tokens to learn the same relative affinity patterns as the teacher's, preserving local geometric structure.

The final semantic pretraining objective for our student encoder f_θ is a weighted sum of these components:

$$\mathcal{L}_{\text{RIDA}} := \lambda_{\text{g}} \mathcal{L}_{\text{global}} + \lambda_{\text{r}} \mathcal{L}_{\text{rank}} + \lambda_{\text{s}} \mathcal{L}_{\text{spatial}}. \tag{5}$$

We use $\lambda_{\text{global}} = 1.0$, $\lambda_{\text{rank}} = 1.0$, and $\lambda_{\text{spatial}} = 0.5$ in our experiments. The resulting network f_θ provides a semantically-structured 3D latent space, which we can now leverage as a powerful loss function for our generative task.

5. Extending LoST to other 3D Representations

Our LoST framework and RIDA objective are designed to be representation-agnostic. Here, we showcase LoST in the TRELLIS [11] latent space. We operate on TRELLIS Stage-1 latents by reshaping the 16^3 voxel grid (feature dimension 8) into a 64^2 2D grid, which preserves our original architecture with 16-dimensional register tokens similar to our adaptation of Direct3D's triplanes. Our method produces variable-length representations that are decoded via TRELLIS Stage-2 to recover high-frequency details, similar to ShapeLLM-Omni [12], while supporting a flexible token budget. We present qualitative results and quantitative results for tokenization on the Objaverse dataset [2] in Figure 1 and Table 3 respectively. TRELLIS additionally models texture compared to Direct3D, which focuses on geometry alone. These results validate the generalization and flexibility of our method.

Figure 1. **LoST in the TRELLIS 3D VAE latent space**. LoST applied to TRELLIS Stage-1 latents demonstrate that the proposed tokenization generalizes beyond the Direct3D triplane representation. These results highlight the flexibility of LoST as a representation-agnostic tokenizer for variable-length 3D generation.

Table 3. **Tokenization Evaluation on Objaverse.** We evaluated 128 high-quality watertight Objaverse assets (filtered via Step1X-3D). These results are consistent with our results on our evaluation set in the main paper. We note that all evaluation is computed on untextured renderings, which focuses on geometry alone (best results are in bold, second best are underlined).

	Num Tokens	CD ($\times 10^{-2}$)↓	DINO↑
OctGPT	∼219	17.925	0.529
	∼15,031	<u>1.210</u>	0.611
	∼239,004	**0.123**	0.729
VertexRegen	∼3,521	3.612	0.545
	∼3,701	1.593	0.595
	∼8,321	0.625	0.753
LoST (Direct3D)	1	**2.460**	**0.690**
	16	**0.963**	**0.779**
	512	0.385	**0.874**
LoST (Trellis)	1	<u>3.242</u>	<u>0.631</u>
	16	1.351	<u>0.702</u>
	512	<u>0.345</u>	<u>0.801</u>

Table 4. **Comparison of Token Dimensions.** We compare total token dimension cost against other autoregressive methods. We note ShapeLLM-Omni uses 32-dimensional tokens for representation but this increases to 3584 due to LLM usage.

	Num Tokens	Token Dimension	Total
OctGPT [9]	∼ 50,000	1	50,000
Llama-Mesh [8]	∼ 3758	4096	∼ 15,392,768
ShapeLLM-Omni [12]	1024	3584	3,670,016
MeshGPT [6]	1200 - 4800	192	230,400 - 921,600
LoST GPT (ours)	**128**	**32**	**4,096**

6. Further Implementation Details

Evaluation. We render four orthogonal views per shape using Blender with detailed shading. We compute all perceptual metrics for each view and report the averaged results.

Tokenizer Training Details. We train the initial 50 epochs without nested dropout to allow the model to prioritize shape reconstruction using its full capacity, while retaining causal masking throughout. We employ mixed precision training with 'bf16' and utilize Exponential Moving Average (EMA) for model weight updates to stabilize training. While we did not explore learned positional encodings or RoPE [7], incorporating these could potentially yield further performance gains. We use an effective batch size of 256 across 8 GPUs for training LoST.

Text Prompt. We provide the prompt template used in Gemini2.5 Pro [3] to produce prompts used to generate our dataset in the next page. In each API call to Gemini, we only produce 500 prompts at a time to ensure the highest quality.

Text Prompt Template

You are a highly creative and meticulous prompt generator for a cutting-edge text-to-3D diffusion model. Your primary task is to generate **500 unique text prompts**, each describing a **single, distinct, and highly visual 3D object or structured scene element.**

—

Goal and Expansive Diversity Constraints:

The generated collection of objects must be **hyper-varied** and **maximally diverse**, drawing inspiration from all forms of media, history, and imagination. Ensure the prompts comprehensively cover the following major categories, with rich, descriptive detail:

1. **Everyday, Tools, and Artifacts:** * **Practical:** A perfectly arranged sushi bento box, a complex wind-up clock mechanism, an antique brass telescope. * **Relics & Treasures:** A glowing Atlantean crystal, a ceremonial Mayan mask, a cursed dagger encrusted with jewels. * Accessories, jewelry, gadgets, household items, musical instruments, sports equipment, clothes, office supplies, toys, etc, etc. Endless possibilties. 2. **Characters, Creatures, and Figurines:** * **Characters from popular culture:**: Examples: Anime characters, Marvel characters such as spider-man, hulk, etc, characters from games such as Ezio Auditore, Lara Croft, Mario, Kratos, cartoon characters, etc., Indian Jones, Sherlock Holmes, Harry Potter, human characters, standard humans such as man, woman, kid, etc. * **Fantasy & Sci-Fi:** Intricate elves, biomechanical cyborgs, ethereal spirits, Lovecraftian monsters. * **Pop Culture & History:** cinematic creatures in dynamic poses, stylized political figures, classic literary characters, characters from animations, sports,. * **Abstract/Stylized:** Chibi characters, low-poly mascots, geometric avatars. 3. **Architecture, Structures, and Scenics:** * **Internal & External:** A collapsing spiral staircase, a sleek Brutalist building facade, an ornate Victorian greenhouse, a subterranean alien throne room, Taj Mahal, Tokyo Tower, Hanging gardens of babylon, sydney opera house. * **Specific Styles:** Hyper-realistic, stylized claymation, cel-shaded, vaporwave aesthetic. 4. **Vehicles and Machinery:** * **Operational & Conceptual:** Detailed vintage motorcycles, futuristic flying battleships, abandoned industrial robots, specialized scientific equipment (e.g., a particle accelerator component). * **Condition:** Rusted, pristine, battle-damaged, overgrown with moss. 5. **Organic, Flora, and Fauna:** * **Animals:** Photorealistic wildlife (e.g., a snow leopard mid-leap), mythical beasts (e.g., a hydra emerging from water), taxidermy displays. * **Plants:** Rare succulents, carnivorous plants, an entire ancient, etc. * Various fruits and vegetables, flowers, trees, fungi, etc. Don't do bonsai, we already have many bonsai prompts. Rather explore diverse things. * Food items and dishes: gourmet dishes, desserts, beverages, noodles, etc.

This is just a guide– use your creativity to explore and expand upon these categories, do not limit yourself to them. ** Choose from absolutely random stuff. Keep a mix of realistic everyday objects/things and creative ones. Focus more on realistic/hyperrealistic. Keep very few futuristic items** —

Formatting and Output Rules:

* Generate **exactly 500** unique, highly visual prompts. **DO NOT REPEAT** any prompt. * The output must contain **only** the sequentially numbered list of prompts. **DO NOT INCLUDE** any introductory text, conversational fillers, or surrounding markdown/code blocks. * The numbering must start at **1.** and proceed sequentially. Ensure **each prompt is on a new line**.

Sample Output (Do NOT repeat these exact prompts): 1. Porcghe 911 Carrera S, hyperrealistic 2. statute of David 3. a sleek, angular neon sign that reads "VOID" 4. an intricate, highly detailed mechanical dragonfly with copper wings 5. a crumbling statue of a griffin perched on a stone pillar ... 500. superhero in a dynamic pose, highly detailed (you can use various superheroes/popular characters/anime characters/game characters)

References

[1] Roman Bachmann, Jesse Allardice, David Mizrahi, Enrico Fini, Oğuzhan Fatih Kar, Elmira Amirloo, Alaaeldin El-Nouby, Amir Zamir, and Afshin Dehghan. Flextok: Resampling images into 1d token sequences of flexible length. In *Forty-second International Conference on Machine Learning*, 2025. 1, 2

[2] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13142–13153, 2023. 3

[3] Google Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. 3

[4] Cijo Jose, Théo Moutakanni, Dahyun Kang, Federico Baldassarre, Timothée Darcet, Hu Xu, Daniel Li, Marc Szafraniec, Michaël Ramamonjisoa, Maxime Oquab, Oriane Siméoni, Huy V. Vo, Patrick Labatut, and Piotr Bojanowski. Dinov2 meets text: A unified framework for image- and pixel-level vision-language alignment, 2024. 1

[5] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3967–3976, 2019. 2

[6] Yawar Siddiqui, Antonio Alliegro, Alexey Artemov, Tatiana Tommasi, Daniele Sirigatti, Vladislav Rosov, Angela Dai, and Matthias Nießner. Meshgpt: Generating triangle meshes with decoder-only transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19615–19625, 2024. 3

[7] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 3

[8] Zhengyi Wang, Jonathan Lorraine, Yikai Wang, Hang Su, Jun Zhu, Sanja Fidler, and Xiaohui Zeng. Llama-mesh: Unifying 3d mesh generation with language models, 2024. 3

[9] Si-Tong Wei, Rui-Huan Wang, Chuan-Zhi Zhou, Baoquan Chen, and Peng-Shuai Wang. Octgpt: Octree-based multiscale autoregressive models for 3d shape generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–11, 2025. 3

[10] Xin Wen, Bingchen Zhao, Ismail Elezi, Jiankang Deng, and Xiaojuan Qi. ” principal components” enable a new language of images. *ICCV*, 2025. 1

[11] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21469–21480, 2025. 3

[12] Junliang Ye, Zhengyi Wang, Ruowen Zhao, Shenghao Xie, and Jun Zhu. Shapellm-omni: A native multimodal llm for 3d generation and understanding. *arXiv preprint arXiv:2506.01853*, 2025. 3

[13] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *The Thirteenth International Conference on Learning Representations*, 2025. 1