STRUCTURED 2D ATLASES VIA GEODESIC OPTIMAL TRANSPORT FOR 3D REPRESENTATION LEARNING

Anonymous authors
Paper under double-blind review

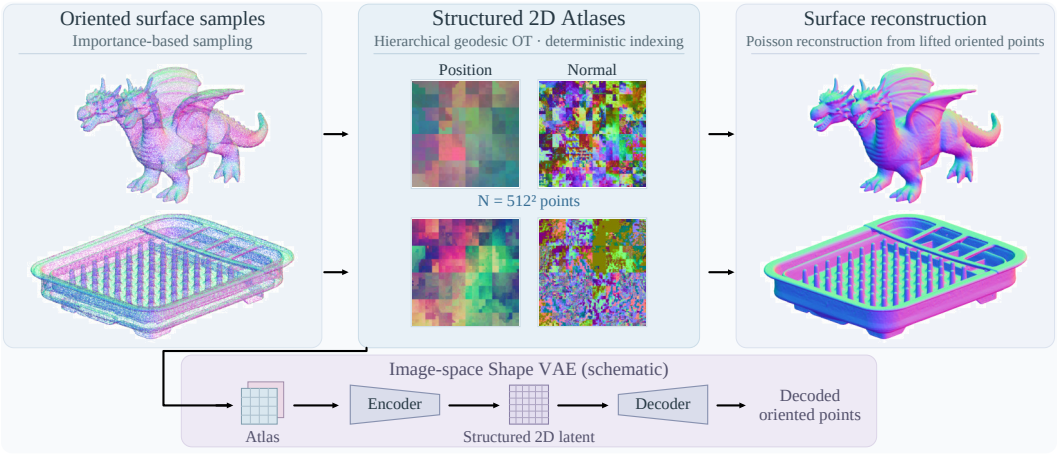

Figure 1: **Structured 2D Atlases for dense 3D representation learning.** Structured position and normal maps organize dense oriented 3D surface samples in a regular 2D domain while retaining their original 3D positions and normals. Compared with the evaluated 3D-native Shape VAEs, our model uses $\mathbf{5.3\times}$ **to** $\mathbf{25.0\times}$ **fewer parameters**, with a trade-off in fine-scale geometric detail.

ABSTRACT

Recent 3D generation pipelines commonly employ Variational Autoencoders (VAEs) to learn compact latent representations of 3D shapes. However, learning high-resolution 3D representations at scale is highly GPU-memory intensive, and existing Shape VAEs often rely on substantial surface subsampling, coarse 3D discretization, or large model capacity to make training tractable. We propose **Structured 2D Atlases**, which reorganize dense oriented 3D surface samples into a regular image-space representation while retaining their original 3D positions and surface normals. Our method establishes a bijection between surface samples and fixed Fibonacci-sphere targets using hierarchical optimal transport guided by intrinsic geodesic distances, followed by deterministic equirectangular indexing into square position and normal maps. An $H \times W$ atlas stores $N = HW$ oriented surface samples, enabling an atlas-based transformer Shape VAE to encode and decode 3D surface geometry through a regular 2D domain rather than a 3D-native representation. Experiments show that our atlas-based Shape VAE uses $\mathbf{5.3\times}$ **to** $\mathbf{25.0\times}$ **fewer parameters** than the evaluated 3D-native Shape VAEs, providing a favorable trade-off between model size and fine-scale geometric detail. These results demonstrate that Structured 2D Atlases provide a parameter-efficient alternative for dense 3D surface representation learning.

1 INTRODUCTION

Generating high-quality 3D assets is a fundamental challenge in computer graphics and computer vision, with applications spanning VR/AR, games, computer-aided design (CAD), physics-based animation, and simulation. Recent advances in direct 3D generation have substantially improved geometric quality and complexity, driven by large-scale 3D datasets and diffusion-based generative

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

models (Zhao et al., 2023; Zhang et al., 2024; Xiang et al., 2024; Zhao et al., 2025; Zhang et al., 2025). Modern pipelines commonly employ a Shape Variational Autoencoder (VAE) to compress 3D geometry into a compact latent representation, followed by a generative model that learns the latent distribution (Zhang et al., 2023; Zhao et al., 2023; Zhang et al., 2024; Chen et al., 2025). However, learning high-resolution 3D representations at scale is highly GPU-memory intensive. Under practical memory constraints, existing Shape VAEs often rely on substantial surface subsampling or discretized 3D representations, while many state-of-the-art models still require large numbers of parameters. Representative formulations include point sets (Charles et al., 2017; Qi et al., 2017; Zhang et al., 2023), voxel-based representations (Ren et al., 2024), and sampled implicit or level-set fields (Park et al., 2019; Mescheder et al., 2019; Osher & Sethian, 1988; Zhao et al., 2023; Zhang et al., 2024; Chen et al., 2025).

This raises a fundamental representation question: can dense 3D surface geometry be organized in a form that enables **substantially more parameter-efficient Shape VAE learning**? Dense surface samples retain direct geometric positions and normals, but their irregular organization makes them expensive to process at scale. Existing Shape VAEs address this through subsampled point sets, sparse or discretized volumetric structures, and implicit fields, often followed by surface extraction such as Marching Cubes (Park et al., 2019; Mescheder et al., 2019; Lorensen & Cline, 1987). However, the geometric information available to the Shape VAE is determined by the representation used during learning, and higher-resolution surface extraction cannot recover information that was not represented during training (Chen et al., 2025). In contrast, **regular 2D representations** provide a structured computational domain well suited to modern image-space architectures, suggesting a more compact way to organize dense surface geometry.

In this work, we introduce **Structured 2D Atlases**, an image-space representation that organizes dense oriented 3D surface samples into regular position and normal maps while retaining their original 3D positions and surface normals. We assign unordered surface samples to fixed Fibonacci-sphere targets using hierarchical optimal transport guided by intrinsic geodesic distances, followed by deterministic equirectangular indexing into a common image-space structure. The fixed target lattice provides consistent pixel addresses across shapes without requiring semantic correspondence. An $H \times W$ atlas stores $N = HW$ oriented surface samples and serves as the native input and output representation of an atlas-based transformer Shape VAE. Experiments show that our Shape VAE uses $\mathbf{5.3\times}$ **to** $\mathbf{25.0\times}$ **fewer parameters** than the evaluated 3D-native Shape VAEs. Quantitative and qualitative results further characterize the resulting trade-off, with the compact model retaining the overall 3D surface structure while exhibiting some smoothing of fine-scale geometric details.

In summary, our contributions are: (i) We introduce **Structured 2D Atlases**, a regular image-space representation for dense oriented 3D surface samples that retains their original positions and normals. (ii) We leverage hierarchical geodesic optimal transport and fixed Fibonacci-sphere targets to construct deterministic position and normal maps with consistent image-space indexing across shapes. (iii) We demonstrate substantially more parameter-efficient Shape VAE learning, with our atlas-based Shape VAE using $\mathbf{5.3\times}$ **to** $\mathbf{25.0\times}$ **fewer parameters** than the evaluated 3D-native Shape VAEs, while characterizing the trade-off in fine-scale geometric detail.

2 Related Work

Shape VAEs and 3D Generative Representations. 3DShape2VecSet (Zhang et al., 2023), Michelangelo (Zhao et al., 2023), CLAY (Zhang et al., 2024), and Hunyuan3D 2.0 (Zhao et al., 2025) learn compact latent representations of 3D geometry, while TRELLIS (Xiang et al., 2024) uses sparse structured 3D latents. These methods support large-scale 3D generation through sampled points, sparse volumetric structures, or implicit representations, and Dora (Chen et al., 2025) further shows that surface sampling can affect reconstruction quality. Our work explores a different representation design by organizing dense oriented surface samples into a regular image-space structure that serves directly as the input and output of a Shape VAE. This provides a more parameter-efficient alternative to conventional 3D-native representations.

Surface and Atlas Representations. FoldingNet (Yang et al., 2018) decodes point clouds from a 2D grid, AtlasNet (Groueix et al., 2018) represents surfaces with learned patches, and Geometry Images (Gu et al., 2002) parameterize 3D surfaces on regular image domains. Gaussian Atlas (Xiang et al., 2025) organizes 3D Gaussian primitives on dense 2D grids, while SPGen (Zhang et al.,

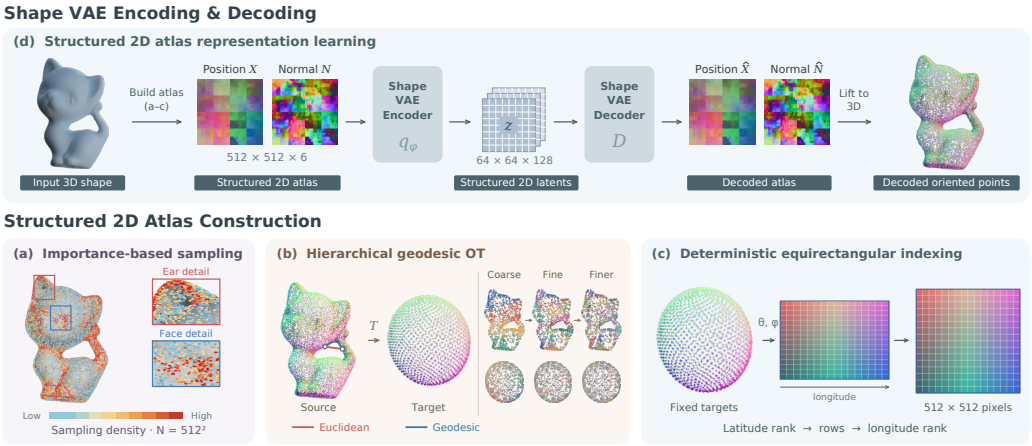

Figure 2: **Overview of Structured 2D Atlases.** (a) Importance-based sampling produces $N = 512^2$ oriented points. (b) Hierarchical geodesic OT maps them bijectively to fixed Fibonacci targets shared across all shapes. (c) Deterministic equirectangular indexing assigns each target a unique pixel in a 512×512 layout shared across shapes. (d) A Swin Transformer-based Shape VAE compresses position–normal atlases into structured 2D latents and reconstructs oriented points.

2025) uses multi-layer spherical projections. These works demonstrate the usefulness of 2D and spherical domains for representing 3D geometry. Our distinction lies in constructing a fixed image-space organization specifically for Shape VAE learning. Unordered oriented surface samples are bijectively assigned to fixed Fibonacci-sphere targets using geodesic-guided hierarchical optimal transport and deterministically indexed into position and normal maps. The resulting atlas retains the original 3D attributes and provides consistent pixel addresses across shapes without assuming semantic correspondence.

Optimal Transport for Surface Mapping. Optimal transport (OT) is widely used for geometric matching and correspondence (Cuturi, 2013; Solomon et al., 2014; Peyré & Cuturi, 2020), while spherical OT has been studied for intrinsic surface parameterization and mapping (Cui et al., 2019; Gotsman et al., 2003), including formulations based on squared geodesic costs (Hamfeldt & Turnquist, 2021). We leverage hierarchical OT guided by intrinsic geodesic distances to establish a bijection between unordered surface samples and fixed spherical targets. This correspondence provides the structured and deterministic organization required to construct our 2D atlas for subsequent Shape VAE learning.

3D Surface Reconstruction. DeepSDF (Park et al., 2019) and Occupancy Networks (Mescheder et al., 2019) recover surfaces from scalar fields using Marching Cubes (Lorensen & Cline, 1987), while DMTet (Shen et al., 2021) and FlexiCubes (Shen et al., 2023) use differentiable isosurface extraction. These approaches reconstruct surfaces through intermediate 3D field representations. Our Shape VAE instead natively predicts structured position and normal maps that map directly back to oriented 3D surface samples. When an explicit mesh is required, the decoded samples are processed using Poisson reconstruction (Kazhdan et al., 2006; Kazhdan & Hoppe, 2013), leaving mesh extraction as a downstream step rather than part of the learned representation.

3 Method

In this section, we present **Structured 2D Atlases** for organizing dense 3D surface geometry in regular 2D image space. We first leverage hierarchical geodesic OT to assign surface samples to fixed Fibonacci targets (Section 3.1). We then describe importance-based oriented sampling and deterministic spherical indexing, which organize the sampled positions and normals into square atlas maps (Sections 3.2-3.3). Finally, a transformer-based Shape VAE learns compact 2D latents and reconstructs dense oriented points from the decoded atlas (Section 3.4). Figure 2 summarizes the pipeline, and Figure 3 illustrates the geodesic distances, assignments, and pixel indexing.

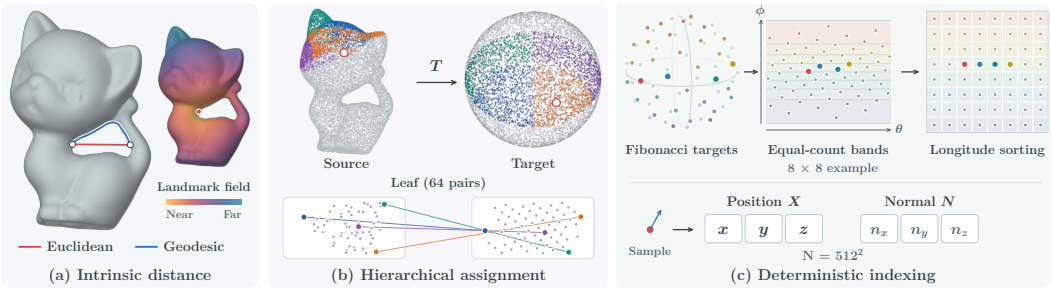

Figure 3: **Hierarchical geodesic transport and atlas construction.** (a) Mesh-edge geodesic distances follow the surface and avoid Euclidean shortcuts through empty space. (b) Hierarchical transport progressively assigns surface samples to fixed Fibonacci-sphere targets, ending with one-to-one assignments within each leaf. (c) The fixed spherical targets are deterministically indexed into unique atlas pixels, where the corresponding 3D positions and normals are stored. An 8×8 example is shown for visualization; the model uses $N = 512^2$ samples.

3.1 Hierarchical Geodesic Optimal Transport

Transport domains. Given a triangular surface $\mathcal{M}$, we center it, rotate it to a deterministic PCA frame, and scale it to unit radius. We sample $\mathcal{P} = \{(\mathbf{x}_i, \mathbf{n}_i)\}_{i=1}^{N}$, where $N = HW = 512^2$. The source and target empirical measures are $\mu_{\mathcal{M}} = N^{-1}\sum_i \delta_{\mathbf{x}_i}$ and $\nu_{\mathbb{S}^2} = N^{-1}\sum_j \delta_{\mathbf{t}_j}$, where $\{\mathbf{t}_j\} \subset \mathbb{S}^2$ is a fixed Fibonacci set. We seek a permutation $\tau : \{1, \dots, N\} \to \{1, \dots, N\}$ assigning source sample i to target $\tau(i)$ while reducing intrinsic-distance distortion. Its inverse $m = \tau^{-1}$ gives the source sample stored at target j.

Surface geodesic distance. Let $G_{\mathcal{M}} = (V, E, w)$ be the mesh-edge graph with $w_{uv} = \|\mathbf{v}_u - \mathbf{v}_v\|_2$. For samples $\mathbf{x}_i \in f_i$ and $\mathbf{x}_{i'} \in f_{i'}$, we use

$$\begin{aligned} d_{\text{edge}}(\mathbf{v}_a, \mathbf{v}_b) &= \min_{(v_0=a,\dots,v_L=b)} \sum_{r=0}^{L-1} w_{v_r v_{r+1}}, \\ d_G(\mathbf{x}_i, \mathbf{x}_{i'}) &= \min_{\substack{u \in V(f_i) \\ v \in V(f_{i'})}} \|\mathbf{x}_i - \mathbf{v}_u\|_2 + d_{\text{edge}}(\mathbf{v}_u, \mathbf{v}_v) + \|\mathbf{v}_v - \mathbf{x}_{i'}\|_2. \end{aligned} \tag{1}$$

Only the endpoint segments lie inside the incident faces, while the connecting path follows the surface graph. Thus, within each connected component, d_G approximates the surface geodesic and avoids Euclidean shortcuts between spatially close but geodesically distant regions. For disconnected components, the virtual bridges described in Appendix A.2 make the graph traversable. Figure 3(a) contrasts the Euclidean and geodesic paths and visualizes a landmark-distance field.

Full geodesic GW objective. We relate the intrinsic surface metric to the spherical great-circle metric $d_{\mathbb{S}^2}(\mathbf{s}, \mathbf{t}) = \arccos(\mathbf{s}^\top \mathbf{t})$ through the squared-loss Gromov-Wasserstein objective

$$\min_{\Gamma \in \mathcal{U}(\mathbf{a},\mathbf{b})} \sum_{i,i',j,j'} \left[\frac{d_G(\mathbf{x}_i, \mathbf{x}_{i'})}{D_M} - \frac{d_{\mathbb{S}^2}(\mathbf{t}_j, \mathbf{t}_{j'})}{D_S} \right]^2 \Gamma_{ij}\Gamma_{i'j'}, \quad \mathbf{a} = \mathbf{b} = \tfrac{1}{N}\mathbf{1}. \tag{2}$$

Here $\mathcal{U}(\mathbf{a}, \mathbf{b}) = \{\Gamma \geq 0 : \Gamma\mathbf{1} = \mathbf{a}, \Gamma^\top\mathbf{1} = \mathbf{b}\}$, and D_M, D_S normalize the surface and spherical distances. This objective specifies the desired intrinsic distortion, but we do not instantiate dense $N \times N$ distance matrices.

Scalable hierarchical approximation. We select $K = 16$ farthest-point landmarks $\{\boldsymbol{\ell}_k\}$ on the surface and $\{\mathbf{u}_k\}$ on the sphere. An orientation-regularized entropic GW problem followed by a $K \times K$ linear assignment yields the landmark correspondence $\sigma(k)$. We evaluate Equation 1 only from each sample to these landmarks and define

$$s_i^{(k)} = \frac{d_G(\mathbf{x}_i, \boldsymbol{\ell}_k)}{D_M}, \qquad r_j^{(k)} = \frac{d_{\mathbb{S}^2}(\mathbf{t}_j, \mathbf{u}_{\sigma(k)})}{D_S}, \qquad c_{ij}^{\text{sig}} = K^{-1}\|\mathbf{s}_i - \mathbf{r}_j\|_2^2. \tag{3}$$

The signature distance c_{ij}^{sig} forms the principal source-to-target cost. A weak directional regularizer disambiguates near-symmetric matches, as detailed in Appendix A.2. We recursively divide source

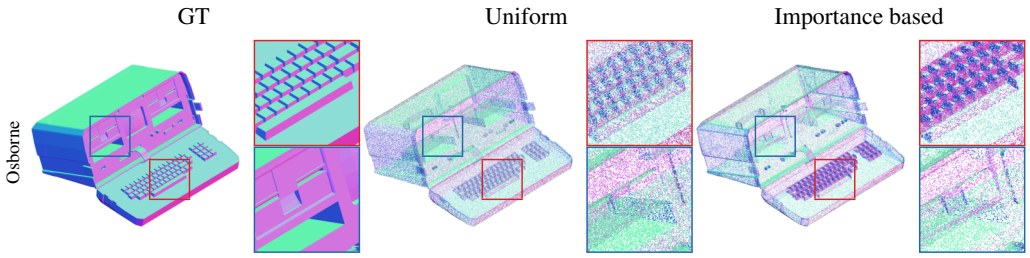

Figure 4: **Importance-based oriented sampling for atlas construction.** (a) Ground-truth surface. (b) Uniform area sampling. (c) Importance-based sampling reallocates the same $N = 512^2$ sample budget toward fine geometric regions while maintaining broad surface coverage. This sampling is applied before atlas organization and Shape VAE learning.

signatures into four equal-cardinality groups and assign them to corresponding atlas regions until 8×8 target blocks. A 64×64 linear assignment resolves each leaf. The disjoint equal-size groups and one-to-one leaf assignments make τ a permutation, so every source sample and target is used exactly once. Figure 3(b) illustrates the hierarchical assignment.

3.2 Importance-Based Oriented Sampling

Before geodesic OT, importance-based sampling (Figure 2(a)) allocates the fixed sample budget between broad surface coverage and fine geometric regions. For face f with area A_f and unsigned dihedral-density κ_f, define $h_f = \min(\kappa_f/\kappa_{\text{ref}}, 1)^\gamma$. We sample faces by

$$p_f = \begin{cases} (1-\alpha)\dfrac{A_f}{\sum_g A_g} + \alpha\dfrac{A_f h_f}{\sum_g A_g h_g}, & \text{if } \sum_g A_g h_g > 0, \\ \dfrac{A_f}{\sum_g A_g}, & \text{otherwise,} \end{cases} \qquad \alpha = 0.5, \quad \gamma = 1. \tag{4}$$

The area term preserves coverage of smooth regions, while the importance term allocates more samples to fine features, sharp edges, and thin structures. Uniform-in-area sampling within face f gives $\rho_{\mathcal{M}}(\mathbf{x}) = p_f/A_f$. Each position is paired with a consistently oriented face normal. Additional details are provided in Appendix A.3.

3.3 Deterministic Equal-Mass Spherical Indexing

The geodesic OT map provides a spherical ordering, which we convert into unique pixel addresses. The Fibonacci construction in Equation 16 provides exactly $N = HW$ targets. For longitude $\theta_j = \text{atan2}(t_{jy}, t_{jx})$ and latitude $\phi_j = \arcsin(t_{jz})$, let $r_j = \text{rank}_\phi(j)$ denote the ascending latitude rank. The continuous equirectangular coordinates and discrete addresses are

$$(\bar{u}_j, \bar{v}_j) = \left(\tfrac{\theta_j+\pi}{2\pi}, \tfrac{\phi_j+\pi/2}{\pi}\right), \qquad (v_j, u_j) = \left(\left\lfloor \tfrac{r_j}{W} \right\rfloor, \text{rank}_{\theta|v_j}(j)\right). \tag{5}$$

The continuous coordinates describe spherical location, while the rank-based construction assigns exactly W targets to each row and uses every pixel address once.

With $F[v_j, u_j] = \mathbf{t}_j$, define

$$R[v_j, u_j] = \mathbf{x}_{m(j)} - \mathbf{t}_j, \qquad \mathbf{N}[v_j, u_j] = \mathbf{n}_{m(j)}, \qquad A = (X, \mathbf{N}), \quad X = F + R. \tag{6}$$

Thus $A \in \mathbb{R}^{H\times W\times 6}$ retains all sampled 3D positions and normals. The spherical lattice determines their image-space order rather than their geometric values, so atlas construction is a permutation without interpolation or sample omission. The target lattice and pixel addresses are fixed across shapes, while atlas values remain shape dependent. This common 2D structure enables compact Shape VAE learning and pixelwise supervision without assuming semantic correspondence across shapes.

Figure 3(c) illustrates the indexing on an 8×8 example. Our main experiments use 512×512 atlases.

3.4 IMAGE-SPACE SHAPE VAE

The Shape VAE in Figure 2(d) operates directly on the dense six-channel atlas using a compact image-space architecture. Positions encode 3D surface locations, while normals explicitly provide local orientation. All atlas pixels enter the encoder, with compression learned through the latent bottleneck.

Encoder and decoder. Our shifted-window transformer encoder defines

$$q_\varphi(\mathbf{z} \mid A) = \mathcal{N}(\boldsymbol{\mu}_\varphi(A), \mathrm{diag}(\boldsymbol{\sigma}^2_\varphi(A))), \qquad \mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, I), \quad \mathbf{z} \in \mathbb{R}^{64\times64\times128}. \tag{7}$$

No feature skips bypass the bottleneck. The decoder outputs

$$\widehat{X} = F + 2\tanh(D_x(\mathbf{z})), \qquad \widehat{\mathbf{N}} = \mathrm{normalize}(\tanh(D_n(\mathbf{z}))). \tag{8}$$

Reading all pixels yields $\widehat{\mathcal{P}} = \{(\widehat{X}_j, \widehat{\mathbf{N}}_j)\}_{j=1}^N$ directly, without querying an implicit field or extracting an isosurface.

Training objective. Fixed atlas indexing provides paired position and normal supervision. Let $w_j \in [1, w_{\max}]$ be a target-derived local-detail weight defined in Appendix A.4. With $\rho_\varepsilon(e) = \sqrt{e^2 + \varepsilon^2} - \varepsilon$,

$$\mathcal{L}_{\mathrm{pair}} = \frac{1}{3N} \sum_{j,c} w_j \rho_\varepsilon(\widehat{X}_{jc} - X_{jc}) + \frac{\lambda_n}{N} \sum_j w_j (1 - \widehat{\mathbf{N}}_j^\top \mathbf{N}_j). \tag{9}$$

We additionally use bidirectional surface loss $\mathcal{L}_{\mathrm{surf}}$, point-to-plane loss $\mathcal{L}_{\mathrm{plane}}$, and boundary-aware atlas regularization $\mathcal{L}_{\mathrm{reg}}$. The complete objective is

$$\mathcal{L}_{\mathrm{VAE}} = \mathcal{L}_{\mathrm{pair}} + \lambda_s \mathcal{L}_{\mathrm{surf}} + \lambda_p \mathcal{L}_{\mathrm{plane}} + \mathcal{L}_{\mathrm{reg}} + \beta D_{\mathrm{KL}}(q_\varphi(\mathbf{z} \mid A) \| \mathcal{N}(0, I)). \tag{10}$$

Appendix A.4 provides the remaining loss definitions, architecture details, and training settings. Reconstruction is obtained directly from the decoded position and normal maps.

4 EXPERIMENTS AND RESULTS

4.1 EXPERIMENTAL SETUP

Implementation details. After preprocessing and filtering, we train on 174,149 shapes from Objaverse (Deitke et al., 2022), ShapeNetCore (Chang et al., 2015), HSSD (Khanna et al., 2023), ABO (Collins et al., 2022), Thingi10K (Zhou & Jacobson, 2016), and GSO (Downs et al., 2022). Each shape is converted into a 512×512 Structured 2D Atlas containing 262,144 oriented surface samples with position and normal channels. Our Shape VAE contains 33.1M parameters and maps the six-channel atlas to a $64 \times 64 \times 128$ latent representation. We train for 150 epochs using AdamW (Loshchilov & Hutter, 2019) with $\mathrm{lr} = \mathrm{wd} = 10^{-4}$, a five-epoch warm-up, and a global batch size of 56 across eight NVIDIA H100 GPUs. Training uses the atlas representation described in Section 3, with all samples participating in the encoder input and supervision. Unless otherwise specified, we evaluate the epoch-149 checkpoint using oriented surface samples read directly from the decoded atlas, without additional refinement or surface reconstruction. Architecture, loss functions, optimization settings, and additional training details are provided in Appendix A.4.

Evaluation protocol. We use Toys1k as a fully held-out reconstruction benchmark, with none of its 1,000 shapes used for training. For each method, we evaluate 100k points from the prediction and reference in the same extent-two coordinate frame without test-time alignment. We report symmetric Chamfer-L_1 distance (CD-L1) (Fan et al., 2016), 95th-percentile bidirectional Hausdorff distance (HD95) (Huttenlocher et al., 1993), F-score at distance threshold 0.01 (Tatarchenko et al., 2019), and reconstruction success rate. Lower CD-L1 and HD95 indicate better geometric agreement, while higher F-score and success rate are better. Parameter counts include only the Shape VAE components used for geometry reconstruction and exclude upstream feature extractors and generative models for all methods. To account for failed reconstructions, F-score is set to zero for failed cases, while CD-L1 and HD95 are averaged over successful reconstructions only.

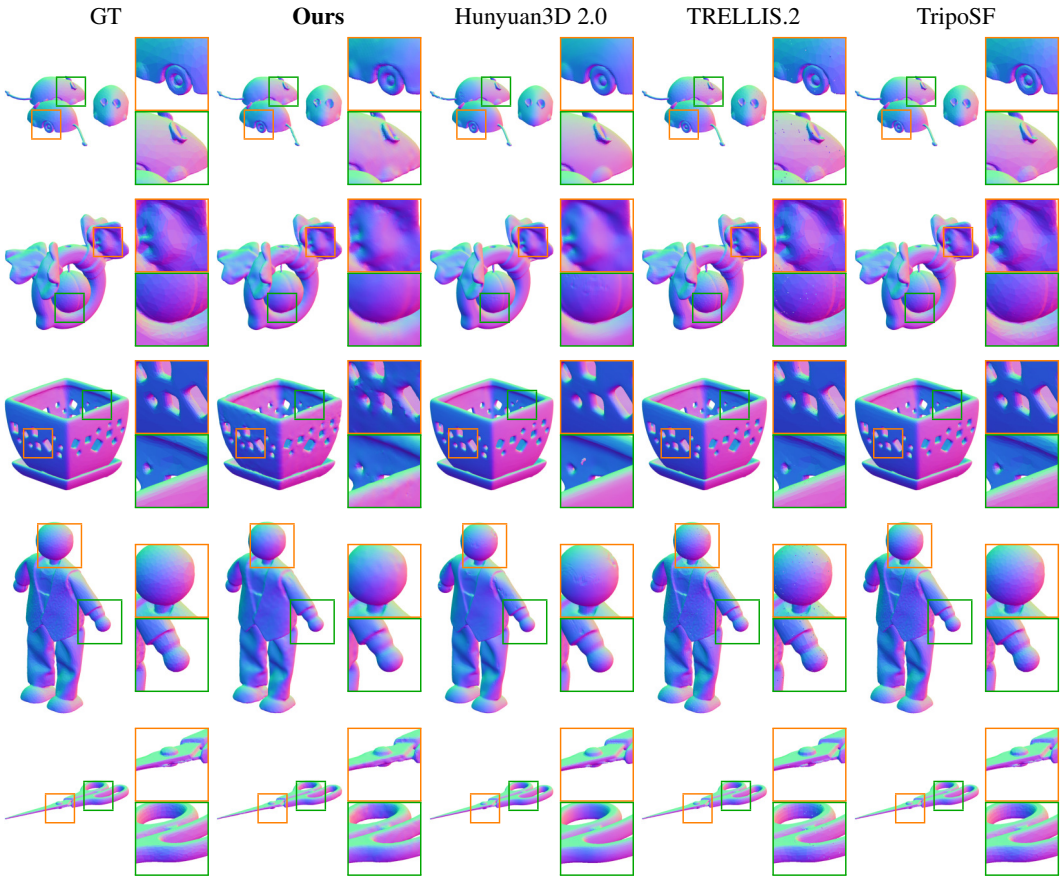

Figure 5: **Qualitative reconstruction comparison.** Matched views and normal maps are used throughout. Our atlas-based Shape VAE uses **5.3× to 25.0× fewer parameters** than the evaluated 3D-native Shape VAEs, illustrating the trade-off between model size and fine-scale geometric detail while retaining the overall 3D surface geometry. Decoded 512×512 oriented samples are refined and converted to surfaces using iterative Poisson reconstruction for visualization.

Table 1: **Comparison of Shape VAE representations.** Existing methods use 3D-native inputs, latents, or decoded fields, whereas ours uses a **Structured 2D Atlas** as the native Shape VAE representation. Surface reconstruction is listed separately from the native decoded output.

Method	Input	Latent	Native Output	Surface Reconstruction
TRELLIS.2 (512^3)	O-Voxels	3D sparse latents	O-Voxels	Flexible Dual Grid
TRELLIS (SLat VAE)	3D voxel features	3D sparse latents	FlexiCubes field	FlexiCubes-based
Hunyuan3D 2.0/2.1	Oriented points	3D tokens	SDF	Marching Cubes
TripoSG	Oriented points	3D tokens	SDF	Marching Cubes
Dora-VAE v1.1	3D points	3D tokens	TSDF	Marching Cubes
TripoSF	3D points	3D sparse latents	Sparse isosurface	FlexiCubes-based
Ours	**Structured 2D Atlas**	**2D atlas latents**	**Oriented points**	**Poisson (optional)**

4.2 Results

Quantitative Results. Tables 1 and 2 assess Structured 2D Atlases as a compact representation for dense 3D surface learning. By encoding and decoding position and normal maps in a regular 2D domain, our atlas-based Shape VAE uses only **33.1M parameters**, corresponding to **5.3× to 25.0× fewer parameters** than the evaluated 3D-native Shape VAEs. On held-out Toys1k, our method ranks third in both CD-L1 and HD95, achieving lower distances than TRELLIS (SLat VAE), Hunyuan3D 2.0/2.1, TripoSG, and Dora-VAE. TRELLIS.2 and TripoSF achieve better distance scores, while several methods obtain higher F-score, indicating a remaining gap in fine-scale geometric

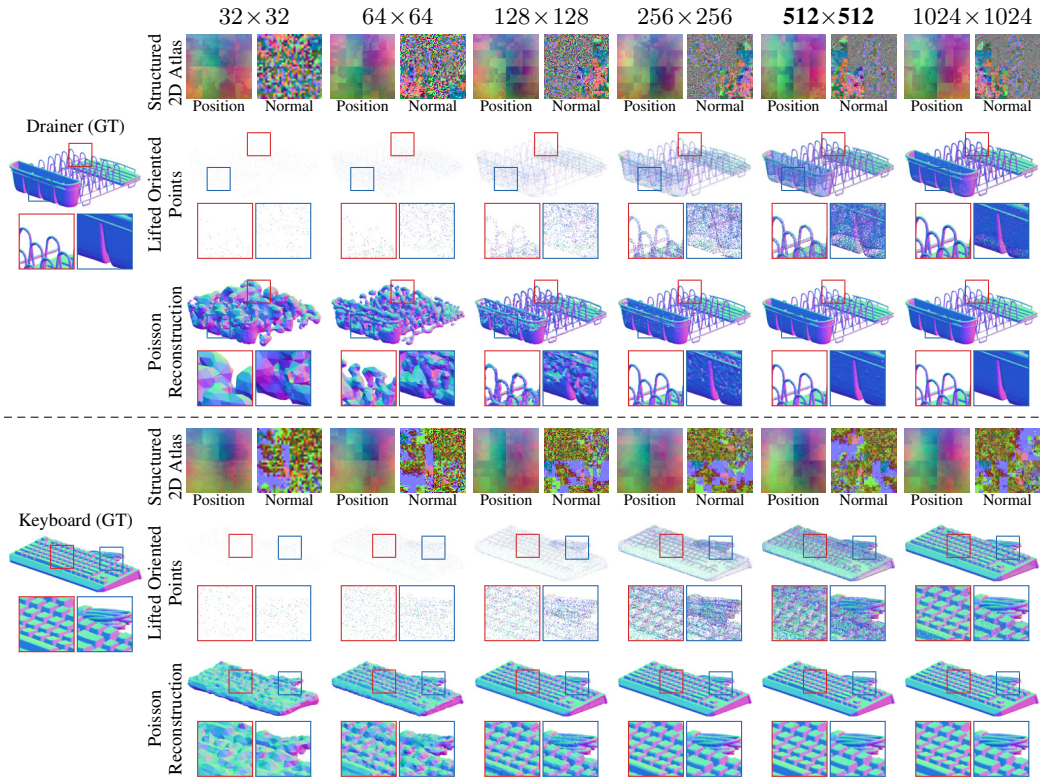

Figure 6: **Effect of atlas resolution on geometric detail.** Structured 2D Atlases store $N = H^2$ ground-truth oriented samples in position and normal maps. Lifted points and matched close-ups illustrate detail coverage as the sample budget increases. This study evaluates atlas construction before VAE compression. Screened Poisson surfaces (fixed depth nine) provide an optional visualization of the represented geometry.

reconstruction. Overall, these results demonstrate a **favorable trade-off between model size and reconstruction quality**: a compact image-space Shape VAE can directly reconstruct dense oriented geometry with substantially fewer parameters, at the cost of some fine-scale geometric detail. The comparison evaluates complete models, as their architectures and latent budgets are not matched.

Qualitative Results. Figure 5 compares our reconstructions with Hunyuan3D 2.0, TRELLIS.2, and TripoSF under matched views and normal maps. Our results retain the overall silhouettes and major geometric structures, while close-ups show smoother ridges and small surface variations in some regions. This illustrates the trade-off between substantially reduced model size and fine-scale geometric detail. The visualized surfaces are obtained from decoded oriented points using refinement and iterative Poisson reconstruction, so their appearance reflects both the learned geometry and post-processing. Table 2 separately evaluates the raw decoded points on held-out Toys1k. Together, these results show that structured position and normal maps can retain broad 3D surface structure while enabling a substantially more parameter-efficient Shape VAE.

4.2.1 Ablation Studies

To characterize our representation, we ablate sample budget and atlas organization before learned compression. Table 3 and Figure 6 show that higher resolution reduces CD-L1 and HD95 on Drainer and Keyboard. F@.01 nearly saturates at 512×512, while 1024×1024 further reduces distances at $4\times$ the samples; DCD is lowest at 256×256. These results support 512×512 as a practical sample-budget trade-off for these examples rather than a universally optimal resolution. At fixed 512^2, importance sampling improves high-curvature coverage, while geodesic OT improves graph-signature locality but not consistently Euclidean neighborhood locality. For fixed samples, ordering changes atlas neighborhoods rather than the lifted point cloud. Appendix A.6 provides these com-

Table 2: **Reconstruction and parameter trade-off on held-out Toys1k.** Our atlas-based Shape VAE directly decodes oriented points using **5.3× to 25.0× fewer parameters** than the evaluated 3D-native Shape VAEs. Metrics use 100k points per shape across 1,000 shapes without alignment; ours uses raw decoded points. CD-L1 and HD95 average successful cases, while F@.01 assigns zero to failures. Parameter counts cover geometry encoder-decoder models only. **Red**, **orange**, and **gold** denote the best, second-best, and third-best results, respectively.

	Parameter Efficiency		Reconstruction			
Method	Params ↓	× Ours	Success (%)	CD-L1 ↓	HD95 ↓	F@.01 ↑
TRELLIS.2 (512^3)	828.6M	25.0×	100.0	**0.0042**	**0.0082**	**95.9**
TRELLIS (SLat VAE)	**176.7M**	5.3×	100.0	0.0094	0.0376	71.7
Hunyuan3D 2.0	327.7M	9.9×	99.5	0.0109	0.0510	83.6
Hunyuan3D 2.1	327.7M	9.9×	99.5	0.0347	0.1955	69.9
TripoSG	242.7M	7.3×	100.0	0.0093	0.0416	**89.6**
Dora-VAE v1.1	191.6M	5.8×	99.4	0.0123	0.0456	54.1
TripoSF	**178.8M**	5.4×	99.3	**0.0043**	**0.0085**	**95.0**
Ours	**33.1M**	**1.0×**	99.8	**0.0085**	**0.0268**	77.1

Table 3: **Ablation on atlas-construction resolution.** Ground-truth atlas points are evaluated without VAE inference or postprocessing. Each resolution uses $N = H^2$ points with area-uniform references, averaged over three sampling seeds. DCD uses $\alpha = 1000$; F is a percentage.

Resolution	Drainer				Keyboard			
	CD-L1 ↓	DCD ↓	HD95 ↓	F@.01 ↑	CD-L1 ↓	DCD ↓	HD95 ↓	F@.01 ↑
32×32	0.037023	0.7452	0.105204	4.21	0.033283	0.7350	0.066418	5.00
64×64	0.018284	0.5624	0.037959	18.61	0.017519	0.5567	0.035941	22.29
128×128	0.009727	0.4651	0.018348	55.58	0.009107	0.4710	0.019546	62.16
256×256	0.005189	0.4372	0.010455	94.53	0.004703	0.4526	0.010206	96.21
512×512	0.002687	0.4406	0.005769	99.98	0.002397	0.4575	0.005305	99.99
1024×1024	0.001361	0.4450	0.002961	100.00	0.001211	0.4627	0.002699	100.00

parisons and hierarchy sensitivity results. Separately, Table 2 shows that our complete Shape VAE uses 5.3–25.0× fewer parameters with the reported reconstruction trade-offs.

5 Conclusion

We introduced **Structured 2D Atlases**, a regular image-space representation for dense 3D surface learning with Shape VAEs. By organizing dense oriented surface samples into structured position and normal maps, our method enables 3D surface geometry to be encoded and decoded through a regular 2D domain while retaining the original 3D attributes. We leverage hierarchical geodesic optimal transport and fixed Fibonacci-sphere targets to construct a deterministic image-space organization that serves directly as the native input and output of an atlas-based transformer Shape VAE. Experiments show that our Shape VAE uses **5.3× to 25.0× fewer parameters** than the evaluated 3D-native Shape VAEs, revealing a substantially more compact operating point for dense 3D representation learning. While this efficiency comes with some smoothing of fine-scale geometric detail, our results suggest that dense 3D surface geometry need not be modeled exclusively through large 3D-native representations. Structured image-space organization offers a promising direction toward smaller and more scalable Shape VAEs for future 3D generative models.

Limitations. Our compact Shape VAE introduces a trade-off between model size and fine-scale geometric detail, with some smoothing in the decoded surface. Hierarchical geodesic OT further trades transport accuracy for scalability by approximating the full-resolution objective. Full pairwise transport requires $\mathcal{O}(N^2)$ time and memory, which is impractical at $N = HW = 512^2$ for large-scale datasets. Our hierarchical construction reduces this to approximately $\mathcal{O}(N \log N)$ time and $\mathcal{O}(N)$ memory, with possible local correspondence errors. Improving geometric detail and transport accuracy while retaining compact model size remains important future work.

AI USE STATEMENT

In this work, we used generative AI tools for paper editing, proofreading, and improving the clarity and presentation of the manuscript. We did not use generative AI tools for developing the research methodology, designing experiments, generating experimental results, or interpreting the results, and the remaining tasks requiring disclosure are not applicable to this work. We have reviewed all AI-assisted work, and all AI-assisted text was reviewed and revised by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide the implementation and evaluation details needed to reproduce our results throughout the main paper and appendix. The experimental setup specifies the training datasets, atlas resolution, Shape VAE architecture, optimization settings, and evaluation protocol. The appendix provides additional details on hierarchical geodesic transport, importance-based sampling, deterministic atlas indexing, loss functions, and model architecture. We also report reconstruction metrics, parameter counts, ablation studies, and qualitative comparisons to support verification of the reported results.

REFERENCES

Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository, 2015. URL https://arxiv.org/abs/1512.03012.

R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 77–85, 2017. doi: 10.1109/CVPR.2017.16.

Rui Chen, Jianfeng Zhang, Yixun Liang, Guan Luo, Weiyu Li, Jiarui Liu, Xiu Li, Xiaoxiao Long, Jiashi Feng, and Ping Tan. Dora: Sampling and benchmarking for 3d shape variational auto-encoders. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 16251–16261, June 2025.

Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F. Yago Vicente, Thomas Dideriksen, Himanshu Arora, Matthieu Guillaumin, and Jitendra Malik. Abo: Dataset and benchmarks for real-world 3d object understanding, 2022. URL https://arxiv.org/abs/2110.06199.

Li Cui, Xin Qi, Chengfeng Wen, Na Lei, Xinyuan Li, Min Zhang, and Xianfeng Gu. Spherical optimal transportation. *Computer-Aided Design*, 115:181–193, 2019. ISSN 0010-4485. doi: https://doi.org/10.1016/j.cad.2019.05.024. URL https://www.sciencedirect.com/science/article/pii/S0010448519302003.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper_files/paper/2013/file/af21d0c97db2e27e13572cbf59eb343d-Paper.pdf.

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, 2022. URL https://arxiv.org/abs/2212.08051.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items, 2022. URL https://arxiv.org/abs/2204.11918.

Haoqiang Fan, Hao Su, and Leonidas Guibas. A point set generation network for 3d object reconstruction from a single image, 2016. URL https://arxiv.org/abs/1612.00603.

Craig Gotsman, Xianfeng Gu, and Alla Sheffer. Fundamentals of spherical parameterization for 3d meshes. *ACM Trans. Graph.*, 22(3):358–363, July 2003. ISSN 0730-0301. doi: 10.1145/882262.882276. URL https://doi.org/10.1145/882262.882276.

Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. Atlasnet: A papier-mâché approach to learning 3d surface generation, 2018. URL https://arxiv.org/abs/1802.05384.

Xianfeng Gu, Steven J. Gortler, and Hugues Hoppe. Geometry images. *ACM Trans. Graph.*, 21(3):355–361, July 2002. ISSN 0730-0301. doi: 10.1145/566654.566589. URL https://doi.org/10.1145/566654.566589.

Brittany Froese Hamfeldt and Axel G. R. Turnquist. A convergent finite difference method for optimal transport on the sphere, 2021. URL https://arxiv.org/abs/2105.03500.

D.P. Huttenlocher, G.A. Klanderman, and W.J. Rucklidge. Comparing images using the hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863, 1993. doi: 10.1109/34.232073.

Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Trans. Graph.*, 32(3), July 2013. ISSN 0730-0301. doi: 10.1145/2487228.2487237. URL https://doi.org/10.1145/2487228.2487237.

Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, SGP '06, pp. 61–70, Goslar, DEU, 2006. Eurographics Association. ISBN 3905673363.

Mukul Khanna, Yongsen Mao, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X. Chang, and Manolis Savva. Habitat synthetic scenes dataset (hssd-200): An analysis of 3d scene scale and realism tradeoffs for objectgoal navigation, 2023. URL https://arxiv.org/abs/2306.11290.

William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '87, pp. 163–169, New York, NY, USA, 1987. Association for Computing Machinery. ISBN 0897912276. doi: 10.1145/37401.37422. URL https://doi.org/10.1145/37401.37422.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4455–4465, 2019. doi: 10.1109/CVPR.2019.00459.

Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988. ISSN 0021-9991. doi: https://doi.org/10.1016/0021-9991(88)90002-2. URL https://www.sciencedirect.com/science/article/pii/0021999188900022.

Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 165–174, 2019. doi: 10.1109/CVPR.2019.00025.

Gabriel Peyré and Marco Cuturi. Computational optimal transport, 2020. URL https://arxiv.org/abs/1803.00567.

Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017.

Xuanchi Ren, Jiahui Huang, Xiaohui Zeng, Ken Museth, Sanja Fidler, and Francis Williams. Xcube: Large-scale 3d generative modeling using sparse voxel hierarchies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Tianchang Shen, Jacob Munkberg, Jon Hasselgren, Kangxue Yin, Zian Wang, Wenzheng Chen, Zan Gojcic, Sanja Fidler, Nicholas Sharp, and Jun Gao. Flexible isosurface extraction for gradient-based mesh optimization. *ACM Trans. Graph.*, 42(4), jul 2023. ISSN 0730-0301. doi: 10.1145/3592430. URL https://doi.org/10.1145/3592430.

Justin Solomon, Raif Rustamov, Leonidas Guibas, and Adrian Butscher. Earth mover's distances on discrete surfaces. *ACM Trans. Graph.*, 33(4), July 2014. ISSN 0730-0301. doi: 10.1145/2601097.2601175. URL https://doi.org/10.1145/2601097.2601175.

Maxim Tatarchenko, Stephan R. Richter, René Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox. What do single-view 3d reconstruction networks learn?, 2019. URL https://arxiv.org/abs/1905.03678.

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. *arXiv preprint arXiv:2412.01506*, 2024.

Tiange Xiang, Kai Li, Chengjiang Long, Christian Häne, Peihong Guo, Scott Delp, Ehsan Adeli, and Li Fei-Fei. Repurposing 2d diffusion models with gaussian atlas for 3d generation, 2025. URL https://arxiv.org/abs/2503.15877.

Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation, 2018. URL https://arxiv.org/abs/1712.07262.

Biao Zhang, Jiapeng Tang, Matthias Nießner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Trans. Graph.*, 42(4), jul 2023. ISSN 0730-0301. doi: 10.1145/3592442. URL https://doi.org/10.1145/3592442.

Jingdong Zhang, Weikai Chen, Yuan Liu, Jionghao Wang, Zhengming Yu, Zhuowen Shen, Bo Yang, Wenping Wang, and Xin Li. Spgen: Spherical projection as consistent and flexible representation for single image 3d shape generation. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, SA Conference Papers '25, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400721373. doi: 10.1145/3757377.3763959. URL https://doi.org/10.1145/3757377.3763959.

Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets, 2024. URL https://arxiv.org/abs/2406.13897.

Zibo Zhao, Wen Liu, Xin Chen, Xianfang Zeng, Rui Wang, Pei Cheng, BIN FU, Tao Chen, Gang YU, and Shenghua Gao. Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=xmxgMij3LY.

Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, Huiwen Shi, Sicong Liu, Junta Wu, Yihang Lian, Fan Yang, Ruining Tang, Zebin He, Xinzhou Wang, Jian Liu, Xuhui Zuo, Zhuo Chen, Biwen Lei, Haohan Weng, Jing Xu, Yiling Zhu, Xinhai Liu, Lixin Xu, Changrong Hu, Tianyu Huang, Lifu Wang, Jihong Zhang, Meng Chen, Liang Dong, Yiwen Jia, Yulin Cai, Jiaao Yu, Yixuan Tang, Hao Zhang, Zheng Ye, Peng He, Runzhou Wu, Chao Zhang, Yonghao Tan, Jie Xiao, Yangyu Tao, Jianchen Zhu, Jinbao Xue, Kai Liu, Chongqing Zhao, Xinming Wu, Zhichao Hu, Lei Qin, Jianbing Peng, Zhan Li, Minghui Chen, Xipeng Zhang, Lin Niu, Paige Wang, Yingkai Wang, Haozhao Kuang, Zhongyi Fan, Xu Zheng, Weihao Zhuang, YingPing He, Tian Liu, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, Jingwei Huang, and Chunchao Guo. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation, 2025.

Qingnan Zhou and Alec Jacobson. Thingi10k: A dataset of 10,000 3d-printing models, 2016. URL https://arxiv.org/abs/1605.04797.

A APPENDIX

In this section, we provide the derivations and implementation details of Structured 2D Atlases. We first formalize geodesic OT and establish its bijection and pixel-occupancy guarantees (Section A.1), then detail the landmark-based hierarchical approximation (Section A.2). Next, we describe importance-based sampling and deterministic atlas indexing (Section A.3). Finally, we specify the Shape VAE architecture, objective, and training protocol (Section A.4), followed by the evaluation protocols (Section A.5) and additional ablation results (Section A.6).

A.1 GEODESIC OT FORMULATION AND OCCUPANCY GUARANTEE

The continuous-density view clarifies the two OT domains. Within a mesh face f, the sampler selects the face with probability p_f and samples uniformly by area, so $\rho_{\mathcal{M}}(\mathbf{x}) = p_f/A_f$ for $\mathbf{x} \in f$. Because $\sum_f p_f = 1$, $\int_{\mathcal{M}} \rho_{\mathcal{M}}\, dA = 1$. The ideal spherical target density is $\rho_{\mathbb{S}^2}(\mathbf{t}) = 1/(4\pi)$; the fixed equal-weight Fibonacci targets approximate it. These are spatial densities, whereas $\mu_{\mathcal{M}}$ and $\nu_{\mathbb{S}^2}$ in the main text are discrete measures. Equal sample weights do not mean uniform sampling density on the input surface.

In Equation 2, $\mathcal{U}(\mathbf{a}, \mathbf{b})$ contains nonnegative matrices with row sums $\mathbf{a}$ and column sums $\mathbf{b}$. The squared loss compares pairs of *source* points through d_G with pairs of *target* points through $d_{\mathbb{S}^2}$; it does not require a distance between points in different domains. The landmark signatures in Equation 15 provide a tractable surrogate for this relational comparison, not an algebraically equivalent reduction of full Gromov–Wasserstein optimization.

Restriction to a deterministic map. Write $\bar{d}_M = d_G/D_M$ and $\bar{d}_S = d_{\mathbb{S}^2}/D_S$. Substituting the coupling of a bijection T into the full objective eliminates the target-index sums:

$$\mathcal{E}_{\mathrm{GW}}(\Gamma_\tau) = \frac{1}{N^2} \sum_{i,i'} \big[\bar{d}_M(\mathbf{x}_i, \mathbf{x}_{i'}) - \bar{d}_S(T(\mathbf{x}_i), T(\mathbf{x}_{i'}))\big]^2. \tag{11}$$

This is the exact objective restricted to a particular bijection. Landmark alignment and hierarchical assignment approximate its minimization; they do not bound the final distortion.

Bijection and marginals. Let $m(j)$ denote the source index assigned to target j. At each internal hierarchy node, equal-sized source groups are assigned one-to-one to disjoint target quadrants. At each leaf, the square linear assignment matches every remaining source point to exactly one target. The claim holds at a leaf. If it holds for four disjoint children, the one-to-one assignment of equally sized source groups to their target quadrants preserves it at the parent. Induction to the root shows that m is a permutation of $\{1, \ldots, N\}$. Therefore m and $\tau = m^{-1}$ define a bijection between the indexed source samples and target points, even if two source samples have identical coordinates. The corresponding permutation coupling is

$$(\Gamma_\tau)_{ij} = \frac{1}{N} \mathbf{1}[j = \tau(i)]. \tag{12}$$

Every row and column of this coupling sums to $1/N$. Hence $T_{\#}\mu_{\mathcal{M}} = \nu_{\mathbb{S}^2}$. This proves exact occupancy and transport marginals, *not* that T globally minimizes Equation 2; coarse hierarchy decisions may be suboptimal.

Pixel uniqueness. The $N = HW$ Fibonacci targets are deterministically sorted, divided into H disjoint rows of W points, and ranked once more within each row. Ranking is a permutation, so each of the HW pixel addresses is used exactly once. Coupled with the bijection above, each atlas pixel stores exactly one oriented source sample. Before 16-bit serialization, Equation 6 and $X = F + R$ give $(X[v_j, u_j], \mathbf{N}[v_j, u_j]) = (\mathbf{x}_{m(j)}, \mathbf{n}_{m(j)})$. Since m is a permutation, the atlas contains the same multiset of oriented samples as the input, merely reordered. Quantization can introduce numerical error, but ordering itself does not delete or duplicate samples.

A.2 Geodesic OT and Hierarchical Assignment

We compute mesh-edge lengths after canonicalization and use deterministic farthest-point traversal over mesh vertices to select source landmarks $\{\ell_k\}_{k=1}^{K} \subset V$. Dijkstra distances from each landmark to all mesh vertices form the source signature basis. For a sample in triangle f_i, its landmark distance is approximated by

$$d_G(\mathbf{x}_i, \ell_k) \approx \min_{v \in V(f_i)} \big(d_{\text{edge}}(v, \ell_k) + \|\mathbf{x}_i - v\|_2 \big). \tag{13}$$

The Euclidean term is confined to the source triangle; it is not a shortcut between arbitrary surface points. Figure 3(a) shows a shortest path on the original mesh edges and a landmark field obtained by interpolating vertex distances over the original faces. These illustrate the graph approximation, not an exact continuous geodesic. On meshes with disconnected components, virtual Euclidean bridges make the graph traversable. We record their use because distances crossing a bridge are not intrinsic geodesics; the illustrated path uses no such bridges.

Let $d_{\mathcal{M}}$ denote exact continuous geodesic distance. Within a connected component containing the landmark, distance to it is 1-Lipschitz by the triangle inequality: $|d_{\mathcal{M}}(\mathbf{x}_i, \ell_k) - d_{\mathcal{M}}(\mathbf{x}_{i'}, \ell_k)| \leq d_{\mathcal{M}}(\mathbf{x}_i, \mathbf{x}_{i'})$. Nearby surface points therefore have similar ideal signatures. The converse does not follow from only $K = 16$ landmarks, and the edge-path approximation in Equation 13 adds error. This motivates the signature cost without claiming an isometric embedding or exact GW minimization.

The same number of target landmarks is selected on the Fibonacci sphere using great-circle farthest-point traversal. We align the two small landmark metric spaces with entropically regularized Gromov–Wasserstein geodesic OT. With normalized landmark distance matrices G and S, the small problem is

$$\min_{\gamma \in \mathcal{U}_K} \quad \sum_{k,k',l,l'} (G_{kk'} - S_{ll'})^2 \gamma_{kl} \gamma_{k'l'} + \eta \langle C_{\text{orient}}, \gamma \rangle - \varepsilon H(\gamma), \tag{14}$$

where $\mathcal{U}_K$ has uniform $1/K$ marginals and $H(\gamma) = -\sum_{k,l} \gamma_{kl} \log \gamma_{kl}$. The weak orientation matrix is $(C_{\text{orient}})_{kl} = \frac{1}{2} \max(0, 1 - \widehat{\boldsymbol{\ell}}_k \cdot \mathbf{u}_l)$, with $\widehat{\boldsymbol{\ell}}_k$ the normalized canonical source landmark direction. We alternate a linearized quadratic cost with balanced Sinkhorn updates, then use a $K \times K$ linear assignment on the resulting coupling. This establishes landmark correspondence σ; it is not a solve of Equation 2 at all N points. For source sample i and target j, we compare normalized geodesic signatures using

$$c_{ij}^{G} = \frac{1}{K} \sum_{k} \left[\frac{d_G(\mathbf{x}_i, \ell_k)}{D_M} - \frac{d_{\mathbb{S}^2}(\mathbf{t}_j, \mathbf{u}_{\sigma(k)})}{D_S} \right]^2 + \lambda_{\text{dir}} \frac{d_{\mathbb{S}^2}(\mathbf{q}_i, \mathbf{t}_j)^2}{\pi^2}, \tag{15}$$

where $\mathbf{q}_i$ is a target-signature proxy direction and π is the circle constant. The weak directional term resolves ambiguous alignments. The large point assignment is then approximated by balanced recursion: each source group is split into four equal-cardinality groups along farthest-pair axes in signature space, a 4×4 assignment matches group summaries to target quadrants, and the process repeats to 8×8 leaves. Each leaf solves a 64×64 linear assignment. Consequently the final map is bijective even though it is not the exact minimizer of the dense GW or linear assignment objective. Early quadrant decisions can still produce geometric correspondence errors. A dense $N \times N$ cost at $N = 512^2$ would occupy roughly 550 GB in float64 before solver overhead.

Complexity. Let B be the terminal block size and K the number of geodesic landmarks. After landmark-distance preprocessing, storing and evaluating the K-dimensional signatures requires $O(KN)$ work and memory. Balanced recursive partitioning has depth $O(\log(N/B))$, and the terminal assignments cost $O((N/B)B^3) = O(NB^2)$. With fixed $K = 16$ and $B = 64$, the point-assignment stage therefore scales approximately as $O(N \log N)$ time and $O(N)$ memory. This describes implementation scaling, not an approximation guarantee: hierarchical decisions may still increase transport distortion.

In Figure 3(b), the four equal-size children are recovered from the saved final permutation: each target quadrant determines its source preimage. The panel therefore visualizes the resulting partition, not a replay of intermediate solver decisions. Its enlarged leaf contains 64 actual source–target pairs; four connecting lines are highlighted for legibility.

A.3 Sampling and Atlas Construction

In Equation 4, κ_f is unsigned dihedral density and κ_{ref} its area-weighted robust quantile. We fall back to area sampling if all importance weights vanish. If $\sum_g A_g h_g > 0$, both terms of Equation 4 sum to one; hence so does p_f. For $0 \le \alpha < 1$ and every nondegenerate face, $p_f \ge (1-\alpha) A_f / \sum_g A_g > 0$. Thus importance weighting cannot exclude flat faces entirely, although a finite sample need not hit every face. Normals are oriented consistently across duplicated and adjacent faces before being paired with sampled positions. A fixed random seed and an area-only reference sample determine the canonical frame independently of the importance weights. Each surface sample retains its source triangle and barycentric coordinates for reproducibility.

The Fibonacci sphere provides exactly $N = HW$ target points. For $j = 1, \dots, N$, we construct

$$\begin{aligned} z_j &= 1 - \frac{2(j - \frac{1}{2})}{N}, \qquad \vartheta_j = (j-1)\pi(3 - \sqrt{5}), \\ \mathbf{t}_j &= \big(\sqrt{1 - z_j^2}\cos\vartheta_j,\ \sqrt{1 - z_j^2}\sin\vartheta_j,\ z_j\big). \end{aligned} \tag{16}$$

Because all z_j are distinct, latitude ranking is deterministic; stable longitude ranking fixes the order within each row. With the one-based target index j in Equation 16, the ascending latitude rank is zero-based, $r_j = N - j$. Row $v = \lfloor r_j / W \rfloor$ contains ranks $vW, \dots, (v+1)W - 1$, and the within-row longitude rank gives $u \in \{0, \dots, W-1\}$. Thus row zero is the southernmost row. The exact latitude boundaries are

$$b_v = \arcsin\left(-1 + \frac{2v}{H}\right), \qquad v = 0, \dots, H. \tag{17}$$

Each band (b_v, b_{v+1}) contains exactly W Fibonacci targets; these are equal-count bands, not uniformly spaced latitude bins. Figure 3(c) displays rows increasing upward and applies this rule to a separate 8×8 Fibonacci example. This reduced indexing example is distinct from the 8×8 production leaf in panel (b). Unlike rounding continuous spherical coordinates, this rank-based construction assigns exactly one target to every atlas pixel. If $m(j)$ is the source-sample index assigned to target j, the serialized atlas contains its position offset and normal as in Equation 6. Offsets and normals are stored in 16-bit channels. At loading time, the fixed target image F restores absolute positions, $X = F + R$. The six-channel payload in Figure 3(c) symbolically denotes the retained (x, y, z, n_x, n_y, n_z) values; it is not a numerical reconstruction. Exact sample preservation in Appendix A.1 concerns reordering before serialization. Channel quantization and the learned VAE bottleneck can each introduce reconstruction error. Only F and its pixel addresses are shape-independent; m depends on each input mesh. This repeatability does not remove pole and longitude-seam effects or guarantee semantic correspondence across different shapes.

A.4 Shape VAE Architecture, Objective, and Training

Nonoverlapping 4×4 patches form 128×128 tokens of width 192. Four fine shifted-window Swin blocks use 8×8 windows and six heads; a 2×2 patch merge forms 64×64 tokens of width 384, followed by eight coarse blocks with 16×16 windows and twelve heads. Row and column embeddings retain atlas addresses. The encoder predicts a 128-channel diagonal Gaussian at each coarse location. The decoder applies eight coarse blocks, expands to the fine grid, and applies four fine blocks before producing six output channels. Its spatial latent has shape $64 \times 64 \times 128$.

Sampled surface losses. For stratified point subsets $\widehat{\mathcal{S}} \subset \widehat{X}$ and $\mathcal{S} \subset X$, the bidirectional surface term is

$$\mathcal{L}_{\mathrm{surf}} = \frac{\lambda_{\mathrm{pre}}}{|\widehat{\mathcal{S}}|} \sum_{\widehat{\mathbf{x}} \in \widehat{\mathcal{S}}} \min_{\mathbf{x} \in \mathcal{S}} \|\widehat{\mathbf{x}} - \mathbf{x}\|_2^2 + \frac{\lambda_{\mathrm{cov}}}{|\mathcal{S}|} \sum_{\mathbf{x} \in \mathcal{S}} \min_{\widehat{\mathbf{x}} \in \widehat{\mathcal{S}}} \|\mathbf{x} - \widehat{\mathbf{x}}\|_2^2. \tag{18}$$

Its first direction measures precision and its second measures coverage. The paired point-to-plane loss is $\mathcal{L}_{\mathrm{plane}} = N^{-1} \sum_j [\mathbf{N}_j^\top (\widehat{X}_j - X_j)]^2$.

Geometry regularization. Let $\mathcal{E}_A$ be the four-neighbor atlas edges and let $M_{jk} \in \{0,1\}$ reject pairs separated by a position or normal discontinuity. The masked first-order term is

$$\mathcal{L}_{\text{grad}} = \frac{\sum_{(j,k)\in\mathcal{E}_A} M_{jk} \left\| (\widehat{A}_j - \widehat{A}_k) - (A_j - A_k) \right\|_1}{\sum_{(j,k)\in\mathcal{E}_A} M_{jk}}, \tag{19}$$

and $\mathcal{L}_\Delta$ applies the same mask to the atlas Laplacian. Boundary and seam terms reproduce target discontinuities instead of smoothing across them. The remaining sampled terms are

$$\mathcal{L}_{\text{aux}} = \lambda_{\text{band}}\mathcal{L}_{\text{band}} + \lambda_{\text{cell}}\mathcal{L}_{\text{cell}} + \lambda_{\text{sn}}\mathcal{L}_{\text{spatial-normal}} + \lambda_{\text{bm}}\mathcal{L}_{\text{balanced-match}}. \tag{20}$$

Thus the grouped term in Equation 10 is

$$\mathcal{L}_{\text{reg}} = \lambda_g\mathcal{L}_{\text{grad}} + \lambda_\Delta\mathcal{L}_\Delta + \lambda_t\mathcal{L}_{\text{tail}} + \lambda_b\mathcal{L}_{\text{boundary}} + \lambda_e\mathcal{L}_{\text{seam}} + \mathcal{L}_{\text{aux}}. \tag{21}$$

For the paired loss, local saliency s_j averages normalized position jumps and normal variation over incident atlas edges that pass the continuity mask. Writing $q_{0.9}(s)$ for its per-shape 90th percentile, we use

$$\eta_j = \min\left(\frac{s_j}{q_{0.9}(s)}, 1\right), \qquad w_j = \min\left(4, 1 + 2\eta_j^{0.75}\right). \tag{22}$$

Thus flat regions retain unit weight and reliable high-detail pixels receive at most a bounded additional weight.

The paired loss in Equation 9 uses $\varepsilon = 10^{-3}$ and $\lambda_n = 0.35$. The evaluated model uses $\lambda_g = 0.5$, $\lambda_\Delta = 0.03$, and $\lambda_p = 0.2$. For Equation 18, the outer sampled-surface weight is $\lambda_s = 0.1$, with $\lambda_{\text{pre}} = 1$ and $\lambda_{\text{cov}} = 1.5$. The coordinate-tail term averages the largest 0.5% of paired position errors and has weight 0.2; prediction and coverage surface tails have weight 0.1. The boundary-coordinate weight is 0.75, while coordinate-jump and normal seam penalties each have weight 0.25. The narrow surface-band, cell-geometry, spatial-normal, and balanced matching weights in Equation 20 are respectively 1, 0.05, 0.1, and 0.01; an additional position–normal cell consistency term has weight 0.03. Balanced matching uses 512 predicted and target points, entropic scale 0.02, 15 Sinkhorn iterations, and normal cost weight 0.005. These terms are evaluated on sampled points to bound memory at 512^2 resolution.

The posterior receives a weak $\beta D_{\text{KL}}(q_\varphi(\mathbf{z} \mid A)\|\mathcal{N}(0,I))$ penalty with $\beta = 10^{-6}$. For predicted mean μ_{pc} and log-variance ℓ_{pc} at latent pixel p and channel c, its elementwise Gaussian form is

$$D_{\text{KL}} = \mathbb{E}_{p,c}\left[\tfrac{1}{2}\left(\mu_{pc}^2 + e^{\ell_{pc}} - 1 - \ell_{pc}\right)\right]. \tag{23}$$

The spatial latent and decoder have no direct feature skips, so the output must be reconstructed through the VAE bottleneck.

Optimization protocol. The geometry autoencoder has 33.1M trainable parameters. We train for 150 epochs with AdamW, learning rate and weight decay both 10^{-4}, a five-epoch linear warm-up, and a global batch size of 56 across eight NVIDIA H100 GPUs. All reported reconstruction results use the fixed epoch-149 checkpoint for our model. Toys1k is excluded from model fitting. The decoder natively predicts all 512^2 positions and normals; neither learned refinement nor surface extraction is used in the primary quantitative comparison.

Optional surface recovery. For explicit surface recovery, decoded oriented points define a normal vector field $\mathbf{V}$ on a 512^3 grid. The optional Poisson field is

$$\chi^\star = \arg\min_\chi \int_\Omega \|\nabla\chi - \mathbf{V}\|_2^2 \, d\mathbf{x}, \qquad \Delta\chi^\star = \nabla\cdot\mathbf{V}. \tag{24}$$

For this energy, integration by parts of its first variation gives $\delta E = -2\int_\Omega(\Delta\chi - \nabla\cdot\mathbf{V})\,\delta\chi\,d\mathbf{x}$ under the solver's periodic boundary convention, yielding the stated Poisson equation. The solution is defined up to an additive constant. For decoded positions $\mathbf{p}_i$ and positive confidence weights ω_i, the input-derived isovalue is $c = \sum_i \omega_i\chi(\mathbf{p}_i)/\sum_i\omega_i$; we extract $\{\mathbf{x} : \chi(\mathbf{x}) - c = 0\}$. This level set is invariant to the additive gauge of χ, since shifting χ by a constant shifts c equally. This field is indicator-like rather than a signed-distance field, and accurate normals alone cannot repair misplaced points or guarantee topology.

A.5 Evaluation and Ablation Protocols

Held-out point reconstruction. Toys1k is used as the held-out reconstruction benchmark. For each method, we evaluate 100k predicted and reference points in the same extent-two coordinate frame without post-hoc alignment. For point sets P and Q, symmetric Chamfer-L_1 is

$$\mathrm{CD}_{L_1}(P,Q) = \frac{1}{2|P|}\sum_{p\in P}\min_{q\in Q}\|p-q\|_2 + \frac{1}{2|Q|}\sum_{q\in Q}\min_{p\in P}\|q-p\|_2. \tag{25}$$

HD95 is the maximum of the two directed 95th-percentile nearest-neighbor distances. F@.01 is the harmonic mean of precision and recall at distance threshold 0.01. Failed reconstructions receive zero F-score; CD-L1 and HD95 are averaged over successful cases in Table 2. The resolution ablation additionally reports DCD with squared Euclidean nearest-neighbor distances, multiplicity compensation, and $\alpha = 1000$. Its density sensitivity means that it need not rank sample budgets identically to CD-L1.

Parameter-count scope. Counts include only each method's geometry encoder and decoder; upstream feature extractors and generative models are excluded. Architectures, latent budgets, and native outputs are not matched. The comparison therefore establishes a favorable measured reconstruction–parameter operating point, but does not by itself prove that atlas organization alone causes the entire parameter reduction.

Construction ablation. Section A.6 presents the controlled sampling, ordering, and hierarchy experiments, including their protocols, metrics, and results. These studies evaluate the representation before VAE compression, separately from the held-out learned reconstruction benchmark.

Resolution ablation. For $H \in \{32, 64, 128, 256, 512, 1024\}$, we construct ground-truth atlases with $N = H^2$ samples and compare their lifted oriented points with independently sampled references. Each entry fixes the lifted atlas and averages three independent reference-sampling seeds. Both prediction and reference counts are N, so changing resolution also changes finite-reference sampling error. This experiment separates construction coverage from VAE error and informs the 512^2 operating resolution; it does not claim universal optimality. The accompanying Poisson surfaces use a fixed depth-nine reconstruction only for visualization and are not learned outputs. A learned resolution comparison would require a separately trained VAE at each resolution with controlled latent capacity.

Qualitative scope. GSO is part of the training corpus, so its examples are in-distribution qualitative illustrations rather than held-out evidence. The main quantitative comparison instead uses raw decoded oriented points on Toys1k. Optional denoising and Poisson recovery are labeled as postprocessing and do not support the raw-point metric claims.

A.6 Additional Ablation Results

This section complements the resolution study in Table 3 and Figure 6. We examine where a fixed sample budget is allocated (Table 4), how ordering changes atlas neighborhoods (Table 5 and Figure 7), and the cost of hierarchical assignment (Table 7). Sampling changes the oriented point set; ordering changes its arrangement in the atlas without changing the represented geometry. These construction-level experiments therefore evaluate coverage and atlas neighborhoods before VAE compression, not learned reconstruction gains or the cause of model-size reduction.

Sampling and ordering controls. We evaluate a fixed 20-shape Toys1k diagnostic subset at 512×512, with three construction seeds per shape. No VAE is trained in this study. Area-uniform and importance sampling are crossed with Euclidean and graph-geodesic OT; random ordering is an additional control. Here, *Euclidean OT* uses Euclidean source–landmark distances and a Euclidean source landmark metric, whereas *graph-geodesic OT* uses the mesh-graph distances described in Appendix A.2. Both use the same landmark-signature matching formulation and hierarchical assignment procedure, with common source landmark indices and the same target lattice. Changing

Table 4: Sampling at $N = 512^2$. Reference-to-sample distance ($\times 10^{-3}$) in high-curvature and remaining regions; lower is better. Means and 95% object-bootstrap intervals after averaging three sampling seeds. High-curvature coverage uses 19 Toys1k shapes because one has no eligible region. This is construction coverage, not VAE reconstruction.

Cohort	Sampling	High curvature	Remaining surface
Toys1k diagnostic (20)	Uniform	2.587 [2.017, 3.229]	2.550 [1.983, 3.182]
Toys1k diagnostic (20)	Importance	2.023 [1.494, 2.608]	3.240 [2.437, 4.153]

the source metric also changes the induced landmark alignment. Thus Euclidean OT is a distance-metric control within our constructor, not direct Euclidean point-to-sphere matching. This common-landmark control differs from the graph-geodesic farthest-point landmark selection used in the production constructor. Within each sampling condition, every ordering is an exact permutation of the same prequantization oriented points. The archive also contains graph-geodesic farthest-point landmark variants for both samplers, giving eight conditions per object–seed pair. All 480 dataset-only conditions across 60 object–seed pairs pass the occupancy and artifact integrity checks; the tables below use the common-landmark controls and random ordering. The Toys1k subset is used only for construction diagnostics, not VAE fitting; these diagnostics are not an additional held-out learning benchmark. Complementary visual and hierarchy comparisons use two user-selected GSO shapes: GSO005-2 (Grimlock) and GSO066-1 (Spelling Board). Each uses the same 512^2 protocol with three seeds, adding 48 verified construction conditions. Table 6 reports their per-shape means, separately from the 20-shape Toys1k aggregate. No custom shapes are used in these construction ablations; the resolution study also uses GSO shapes.

A.6.1 Effect of Importance Sampling

Table 4 measures distance from independent area-uniform reference samples to the constructed point set, separately in high-curvature and remaining regions. The high-curvature region is fixed from the source mesh's unsigned edge-dihedral density, using an area-weighted 80th percentile over source faces (retaining positive curvature when the threshold is zero); it is not redefined for each sampler. One diagnostic Toys1k shape has no eligible high-curvature region and is omitted only from that regional statistic. Importance sampling reduces mean distance in high-curvature regions by approximately 22% on the Toys1k diagnostic subset, while increasing it on the remaining surface. Thus it reallocates a fixed sample budget rather than improving uniform coverage everywhere. This supports the sampling mechanism, not a claim of lower learned reconstruction error.

These measurements complement the main-paper sampling illustration in Figure 4. On the two GSO diagnostic shapes, Table 6 shows approximately 21% and 31% lower high-curvature mean distance for Grimlock and Spelling Board, respectively, with increased distance on the remaining surface.

A.6.2 Effect of Atlas Ordering

This experiment changes atlas organization while preserving the underlying oriented point set. For a fixed sampler, let m_o denote the permutation from target indices to source samples under ordering o. Before quantization, lifting any of these atlases gives the same multiset, $\mathcal{P}_o = \{(\mathbf{x}_{m_o(j)}, \mathbf{n}_{m_o(j)})\}_{j=1}^{N} = \mathcal{P}$. Random, Euclidean OT, and geodesic OT therefore have identical point-set CD-L1, HD95, and F-score against a fixed reference when all retained samples are evaluated. Figure 7 therefore compares only their normal atlases, without repeating the identical lifted point clouds. The quantities that change are pixel assignments and the 3D relationships between neighboring pixels; we evaluate these below.

For each horizontal or vertical atlas edge (i, j), we measure $J_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2 / [(d_8(\mathbf{x}_i) + d_8(\mathbf{x}_j))/2]$, where d_8 is the distance to the eighth nonself Euclidean neighbor. J_{95} is its 95th percentile. We also compute $S_{ij} = \max_k |d_G(\mathbf{x}_i, a_k) - d_G(\mathbf{x}_j, a_k)|$ using eight evaluation landmarks a_k excluded from construction, and report its 95th percentile S_{95}. These triangle-entry mesh-graph distances are descriptive proxies; they are not exact continuous geodesics. Table 5 shows much larger discontinuities under random ordering. Geodesic OT has lower S_{95} than Euclidean OT, but its J_{95} is substantially worse on the Toys1k diagnostic subset. Ambient proximity and graph-signature proximity are dif-

Table 5: Ordering diagnostics at 512^2, with the same importance-sampled points. J_{95} measures density-normalized Euclidean jumps; S_{95} measures independent landmark-signature differences. Lower values are better for each diagnostic, but neither guarantees learned reconstruction quality. Mean [95% object-bootstrap interval], after averaging three seeds.

Cohort	Ordering	J_{95}	S_{95}
Toys1k diagnostic (20)	Random	307.044 [261.886, 356.203]	5.462 [4.095, 6.945]
Toys1k diagnostic (20)	Euclidean OT	10.660 [9.529, 11.819]	2.841 [1.809, 3.885]
Toys1k diagnostic (20)	Geodesic OT	43.145 [34.906, 51.692]	0.222 [0.105, 0.418]

Table 6: Construction diagnostics on GSO005-2 (Grimlock) and GSO066-1 (Spelling Board) at $N = 512^2$, averaged over three seeds. Coverage is reference-to-sample mean distance ($\times 10^{-3}$), partitioned into fixed high-curvature (High) and remaining (Rest) regions. J_{95} and S_{95} follow Appendix A.6.2. Lower is better within each metric. These are construction diagnostics, not learned reconstruction results.

Shape	Sampling	Ordering	High	Rest	J_{95}	S_{95}
Grimlock	Uniform	Geodesic OT	2.670	2.816	46.226	0.391
Grimlock	Importance	Random	2.099	3.180	159.364	6.400
Grimlock	Importance	Euclidean OT	2.099	3.180	8.892	1.248
Grimlock	Importance	Geodesic OT	2.099	3.180	50.031	0.391
Spelling Board	Uniform	Geodesic OT	2.511	2.575	11.019	0.097
Spelling Board	Importance	Random	1.729	3.379	317.209	3.035
Spelling Board	Importance	Euclidean OT	1.729	3.379	22.576	0.950
Spelling Board	Importance	Geodesic OT	1.729	3.379	16.360	0.092

ferent criteria, and the hierarchy does not optimize either diagnostic globally. Disconnected or unwelded source meshes can further affect graph distances; virtual bridges are not intrinsic surface paths. These results therefore do not justify a universal locality or learned-accuracy guarantee. Different atlas neighborhoods may affect image-space learning, but these diagnostics alone cannot establish that effect. Testing it requires separately trained VAEs with matched architecture, latent capacity, losses, training data, and optimization budget, followed by evaluation of their decoded points rather than lifted ground-truth samples.

For the selected GSO shapes, Table 6 shows that geodesic OT reduces S_{95} relative to Euclidean OT for both shapes, but reduces J_{95} only for Spelling Board, illustrating the distinction between the two locality criteria. The source meshes undergo the constructor's standard face-validity checks, without additional smoothing or component cleanup. The prepared graphs contain 1,887 components for Grimlock and 12 for Spelling Board, connected by 1,886 and 11 virtual bridges, respectively. Distances crossing these bridges are not intrinsic surface paths. These GSO examples test the constructor on fragmented mesh inputs, rather than establish purely intrinsic locality on connected surfaces. They supplement the broader diagnostic cohort and do not establish dataset-wide generalization or held-out VAE reconstruction performance.

A.6.3 Hierarchical Assignment Sensitivity

We vary the terminal assignment size over 16, 64, and 256 points at 512^2 on GSO005-2 and GSO066-1, with three sampling seeds and otherwise fixed signatures and targets. Every leaf size is freshly solved from these shared inputs; historical cached assignments are not mixed with new solutions. Component bridges are retained consistently across all settings and do not represent intrinsic paths across empty space. A separate 32^2 control permits an exact flat solve of the same linear signature-cost assignment. This control tests the hierarchy's approximation, not a low-resolution replacement for the native-512^2 study or an exact solve of the relational GW objective. Table 7 reports final-pair surrogate costs, neighborhood discontinuities, assignment-only runtime, and the largest leaf cost-matrix size. The latter is an analytical allocation size, not measured total memory. This comparison separates assignment cost, neighborhood behavior, and computational overhead rather than assuming that a larger terminal block improves every criterion. The default 64-point leaf is a computational operating choice, not a claim of optimal geometric locality. All variants pre-

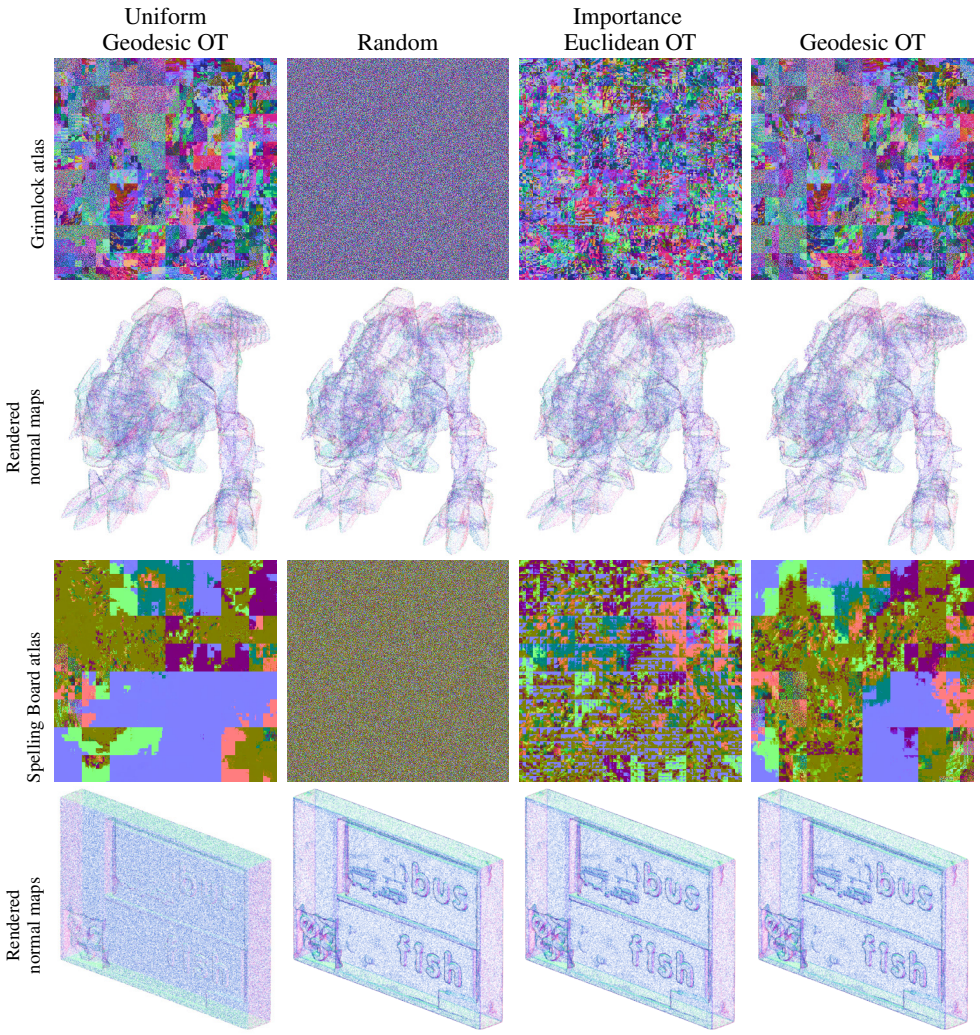

Figure 7: **Sampling and ordering on GSO shapes.** GSO005-2 (Grimlock) and GSO066-1 (Spelling Board). Normal atlases and rendered normal maps use $N = 512^2$ samples (seed 27). The importance-sampled columns reorder the identical oriented point set; their 3D renders are therefore identical. Uniform sampling changes the point set. No VAE or surface reconstruction is used.

serve one sample per pixel. As in the ordering study, changing the leaf size changes the assignment, not the underlying point set; this table measures the assignment's computational and neighborhood trade-offs rather than reconstruction error.

B Extra Qualitative

Table 7: Hierarchical assignment sensitivity on GSO005-2 (Grimlock) and GSO066-1 (Spelling Board), averaged over three sampling seeds. The 32^2 control permits a flat exact solve of the signature-cost assignment, not full GW. Time excludes graph/signature preprocessing; cost-matrix size is analytical, not peak process memory. Every condition has 100% pixel occupancy.

Resolution	Leaf points	Surrogate cost ↓	J_{95} ↓	Time (s)	Leaf matrix (MiB)
512^2	16	0.05677	33.21	2.82	0.002
512^2	64	0.05677	33.20	2.65	0.031
512^2	256	0.05676	33.29	13.77	0.500
32^2	16	0.06359	6.41	0.01	0.002
32^2	64	0.06303	6.55	0.01	0.031
32^2	256	0.05949	6.50	0.04	0.500
32^2	1024 (flat)	0.04771	6.23	0.28	8.000

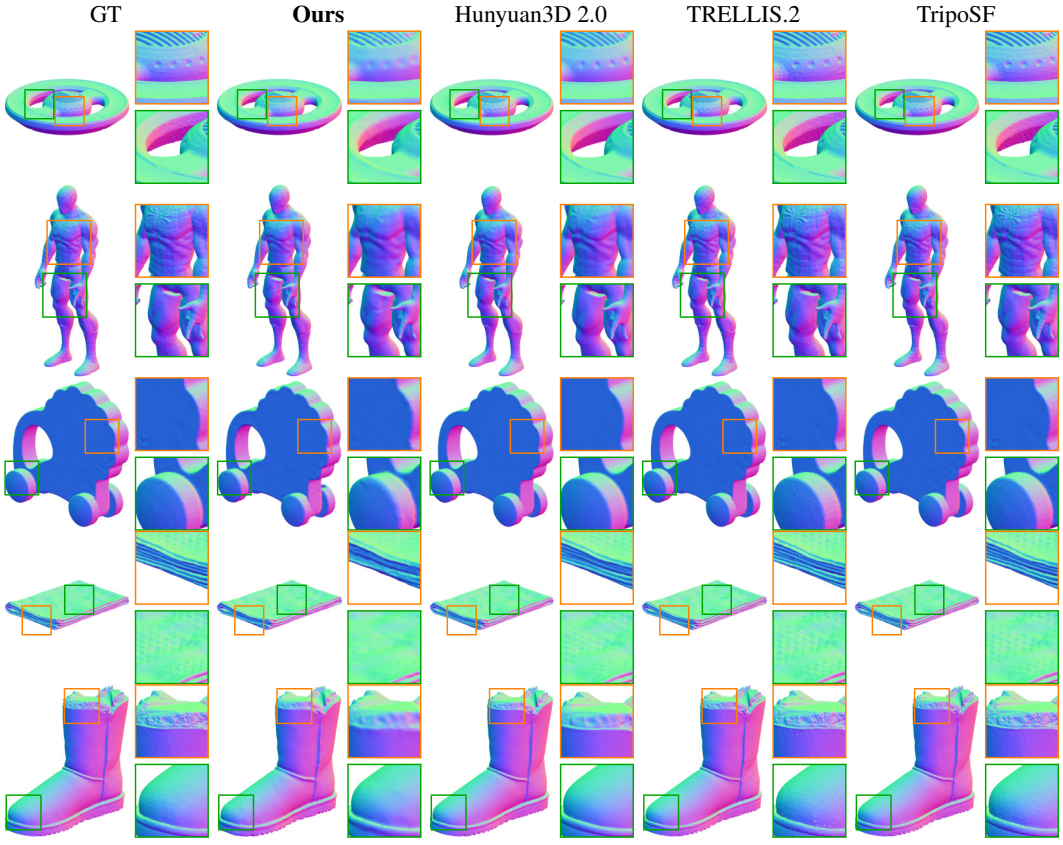

Figure 8: **Qualitative reconstruction on GSO shapes.** We compare ground truth with our method, Hunyuan3D 2.0, TRELLIS.2, and TripoSF under matched viewpoints and geometric-normal rendering. Orange and green insets highlight local geometric details. Ours uses the 512×512 Shape VAE with refinement and iterative Poisson surface reconstruction.

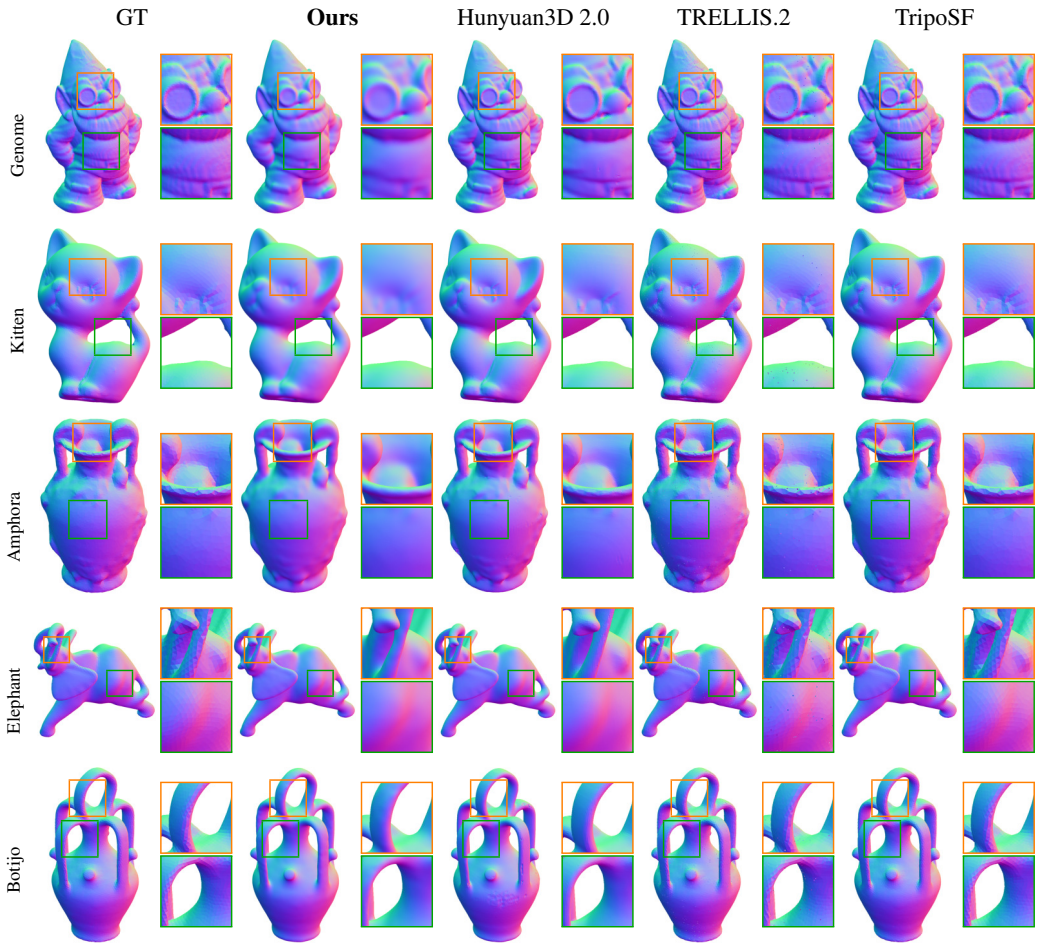

Figure 9: **Qualitative comparison on custom shapes.** We compare ground truth with our method, Hunyuan3D 2.0, TRELLIS.2, and TripoSF under matched viewpoints and geometric-normal rendering. Orange and green insets highlight local geometric details. Ours uses the 512×512 Shape VAE with refinement and Poisson surface reconstruction.