VNC: A Scale-Space Foundation for Learnable 3D Surface Evolution

Baoxing Li, Yong Deng, and Xu Zhao✉

Shanghai Jiao Tong University, Shanghai, China
{lbx11sjtu,dengyong2016,zhaoxu}@sjtu.edu.cn

Abstract. Controlling continuous, physically valid 3D surface evolution is a fundamental challenge in computer vision and graphics. Current neural representations typically rely on unstructured latent spaces or discrete iterations, lacking explicit geometric interpretability and struggling to produce smooth deformations. While classical scale-space theory offers clear physical interpretability, its numerical irregularity isolates it from modern deep learning. In this paper, we bridge this gap by establishing a learnable scale-space foundation. Our framework comprises three synergistic components: (1) the Variational Neighborhood Curvature (VNC) operator, an efficient, parallelizable, and scale-stable geometric metric, (2) the spacetime-balanced VNC-Flow algorithm, which translates irregular physical smoothing into structured deformation trajectories, and (3) a Controllable Variational Autoencoder (C-VAE) that learns these trajectories conditioned on normalized evolution time. Departing from traditional discrete iterations or unstructured latent interpolations, our continuous neural surrogate induces a structured radial latent organization. This enables precise control over continuous shape abstraction and establishes a structural prior for downstream applications.

Keywords: Variational Neighborhood Curvature, 3D Surface Evolution, Curvature Flow, Scale Space, Structured Latent Space

1 Introduction

Recent successes in 3D generative modeling and representation learning [10, 13, 16, 31] have largely been driven by data-driven neural architectures. While these models excel at generating highly detailed complex shapes from scratch, they primarily function as static generators. Controlling the *continuous, geometrically valid evolution* of these surfaces, such as progressively smoothing out fine wrinkles under physical laws while strictly preserving the underlying macroscopic topology, remains a fundamental challenge [8, 19]. In classical computer graphics, this problem is elegantly solved using *scale-space theory* [15]. By applying physical geometric flows like mean curvature flow (MCF) [3, 11], a surface can be viewed through continuous observation scales. Acting as a "smart iron", the flow progressively irons out high-frequency details layer by layer. However, such explicit mathematical computations are computationally heavy, numerically irregular, and largely isolated from modern deep generative models. To bridge this gap, we propose encapsulating the classical scale-space into a *learnable, continuous neural surrogate.*

Conceptually, the evolution of a 3D surface can be understood as the progressive simplification of topologies and details. To drive this evolution, modern deep learning has largely explored two distinct paradigms. First, iterative methods deform explicit meshes [8, 24] or implicit level-sets [19, 22] directly. These methods frequently suffer from unbalanced deformations, topological singularities, and fail to respect the physically-grounded, frequency-ordered decay fundamental to scale space. Second, latent interpolation methods [4, 27] navigate unstructured latent spaces. Unfortunately, linear operations on abstract codes lack local geometric interpretability, typically producing non-geometric artifacts or hallucinated discrete jumps. To achieve physically controllable surface evolution, a paradigm shift is required: *instead of relying on unstable discrete iterations or unstructured latent interpolations, we explicitly learn a continuous, physics-driven scale space via a temporally-conditioned latent mapping.*

However, leveraging classical curvature flow to supervise deep neural networks is highly non-trivial because the physical evolution of a surface under MCF is *numerically irregular and topology-unstable.* First, it suffers from severe spatial imbalance: flat regions may over-simplify into degenerate shapes long before complex, high-frequency wrinkles are fully resolved. Second, it is temporally unpredictable and prone to topological singularities (e.g., pinching or self-intersections), resulting in highly irregular deformation trajectories. Deep generative models, such as Variational Autoencoders (VAEs) [12, 34], require predictable, identically distributed, and topologically stable intermediate states to learn continuous mappings. Traditional explicit MCF fundamentally fails to provide such structured data.

To translate this irregular physical flow into a learnable neural representation, we propose a unified framework (Fig. 1) comprising three synergistic components. First, we introduce the **Variational Neighborhood Curvature (VNC)** operator (Sec. 3.1), a fully differentiable, scale-stable, and highly parallelizable geometric metric that overcomes the computational bottlenecks of classical discrete curvature estimators. Second, guided by VNC, we propose the **spacetime-balanced VNC-Flow** algorithm (Sec. 3.2). By enforcing a strict spatial information equilibrium and a temporal geometric decay budget, it translates the unpredictable physical flow into reliable, singularity-free, stage-wise continuous deformation trajectories. Finally, we introduce a **Controllable VAE (C-VAE)** (Sec. 3.3) conditioned on the normalized evolution time to act as the continuous neural surrogate for the discrete geometric evolution. For versatility, this generative framework is instantiated in two variants: one for dataset-level generalization and another for high-fidelity, instance-specific distillation.

Crucially, learning this neural surrogate encourages the generative latent space to organize into a highly structured ***radial latent structure***. Instead of an unstructured mapping from noise, minimum-information base shapes cluster densely at the center of the space, while complex high-frequency details radiate predictably toward the periphery. Evaluated on highly complex 3D hu-

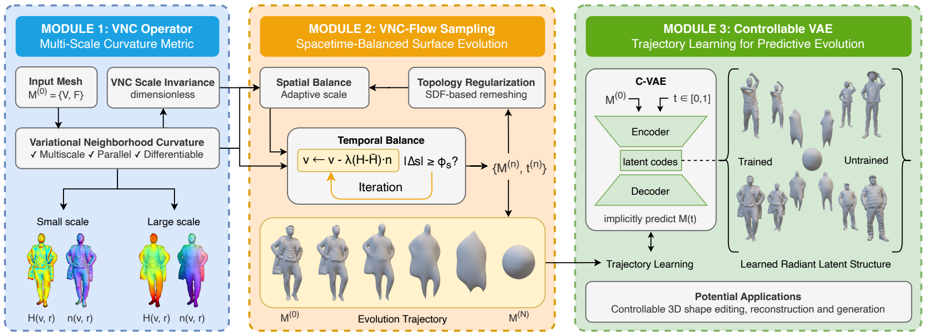


Fig. 1: Pipeline of our learnable 3D surface evolution framework. Left: The VNC operator computes fast, differentiable, and scale-stable multi-scale curvatures. **Middle:** Guided by VNC, the spacetime-balanced VNC-Flow explicitly samples stable, singularity-free deformation trajectories toward a spherical base topology. **Right:** Conditioned on these physical trajectories, the C-VAE inherently organizes a structured radial latent space, enabling continuous predictive evolution and establishing a robust geometric prior for downstream tasks.

man surfaces, our paradigm supports valid geometric interpolation and explicitly interpretable continuous shape abstraction without requiring iterative physical simulations at inference time. In summary, our main contributions are:

– We propose the **VNC operator**, a mathematically dimensionless and highly parallelizable metric whose normalized response is scale-stable, seamlessly integrating multi-scale geometric analysis into deep learning pipelines.
– We design the spacetime-balanced **VNC-Flow**, a physical-to-neural sampling algorithm that successfully transforms irregular curvature flow into temporally and spatially structured training trajectories.
– We develop the time-conditioned **C-VAE** as a continuous neural surrogate. This architecture inherently induces a geometrically interpretable *radial latent structure*, enabling controllable, physics-guided 3D surface evolution.

2 Related Work

Our work bridges classical differential geometry with modern neural generative models. We review literature across three key domains aligned with our method.

2.1 Multi-scale Geometric Analysis

Estimating discrete curvature is foundational for 3D shape analysis [2,21]. While classical methods achieve multi-scale robustness via integral invariants [26] and scale-space theory [15], their reliance on explicit, computationally expensive spatial querying (e.g., line-ball intersections) inherently isolates them from modern GPU-accelerated deep learning. Conversely, recent neural-integrated estimators [18] act purely as static local operators, lacking multi-scale capabilities.

To bridge this gap, we propose the VNC operator. By mathematically reformulating local spatial queries into global Gaussian-weighted matrix multiplications, VNC captures multi-scale geometry seamlessly without explicit spatial indexing. Crucially, VNC normalizes the geometric response to provide a *dimensionless, scale-stable shape index*, establishing a robust multi-scale foundation for physics-driven surface evolution.

2.2 Geometric Flows and Surface Evolution

Surface evolution via mean curvature flow (MCF) is a classical tool for geometry processing [5,11]. Classical level-set theory [23] provided the foundational mathematical framework for tracking topological changes during implicit surface evolution. In the deep learning era, this concept has evolved into two main streams. For explicit surfaces, Neural ODEs drive continuous mesh deformations [8,24]. For implicit representations, early approaches formulated geometry processing on neural fields via invertible deformation networks [32]. Subsequent works deformed level sets using hybrid Eulerian-Lagrangian tracking [19] or space-time coordinate conditioning [22].

Despite these advances, existing neural flows largely follow a *learning-to-deform* paradigm, often ignoring the uneven spatial deformation rates. In contrast, the proposed VNC-Flow algorithm introduces a *spacetime-balanced sampling* paradigm. By constraining the flow toward uniform geometric information decay, it makes the generated deformation sequence more predictable and topologically stable.

2.3 Structured Neural 3D Representations

To represent high-fidelity 3D surfaces, recent methods typically rely on implicit neural fields parameterized by MLPs with periodic activations [28] or Eikonal regularization [7], eventually extracting the explicit mesh via the marching cubes algorithm [17]. Building on these foundational fields, recent state-of-the-art 3D generative models, including large reconstruction models [10,29,31] and 3D diffusion [14,30], leverage expressive geometric backbones [13,34] for unprecedented zero-shot synthesis. Despite their remarkable visual fidelity, these frameworks rely on uninterpretable, unstructured latent spaces that map arbitrary noise directly to complex shapes. This "black-box" mapping makes progressive, physics-governed topological editing nearly impossible.

The proposed framework departs from this unstructured paradigm. Built upon the VecSet encoder [13,34], our C-VAE serves as a continuous neural surrogate trained on well-defined VNC-Flow sequences. By enforcing this temporal conditioning, we transform the conventionally entangled latent space into a highly interpretable *radial latent structure*. Consequently, it establishes a robust physical and structural prior, laying the groundwork for future exploration in controllable and topology-stable 3D generation.

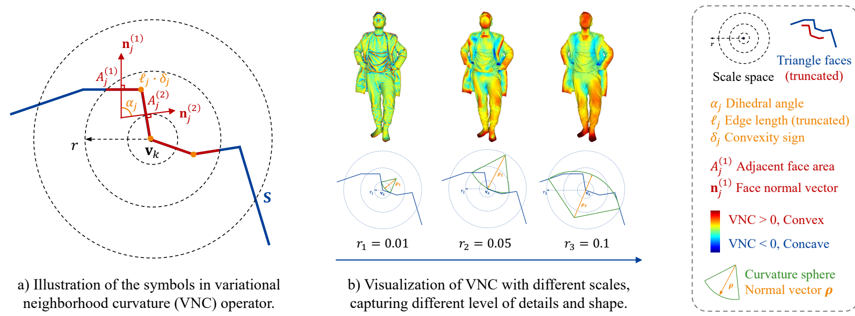


Fig. 2: VNC definition and visualization. (a) VNC-hard explicitly computes curvature via line-ball intersections. (b) As the perception scale increases, perceived curvature naturally transitions from high-frequency details to low-frequency structures.

3 Method

3.1 Variational Neighborhood Curvature Operator

Inspired by the integral invariants framework [26], we introduce the Variational Neighborhood Curvature (VNC), a fully differentiable, multi-scale discrete mean curvature operator. To optimally balance theoretical rigor with the highly parallelized computational demands of deep learning, we design two synergistic variants: the explicitly bounded **VNC-hard** (as a theoretical anchor) and the probabilistically bounded **VNC-soft** (as the practical neural enabler).

VNC-hard (Explicit Boundary). Rooted in classical normal cycle theory [2], as shown in Fig. 2, the *integrated discrete mean curvature* over an explicit spherical neighborhood $\mathcal{N}_r(k)$ for a vertex $\mathbf{v}_k$ is defined as:

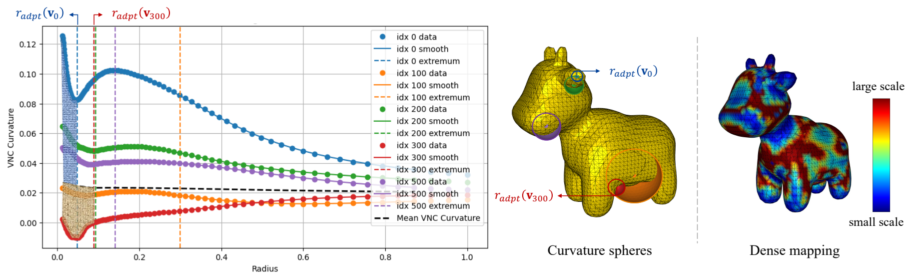
$$H_{\text{int}}(\mathbf{v}_k, r) \triangleq \frac{1}{2} \sum_{j \in \mathcal{N}_r(k)} \alpha_j \cdot \ell_j \cdot \delta_j \tag{1}$$

where $\mathcal{N}_r(k)$ denotes the set of edges geometrically intersecting the ball neighborhood at scale r, ℓ_j is the exact length of the intersected segment of edge e_j, $\alpha_j \in [0, \pi]$ is the dihedral angle between adjacent faces, and $\delta_j \in \{-1, +1\}$ is the convexity sign.

To ensure low-frequency continuity, we extend this formulation by introducing a Gaussian spatial weighting kernel $w_{kj} = \exp\left(-d_j(\mathbf{v}_k)^2/2\sigma^2\right)$, where $d_j(\mathbf{v}_k)$ is the distance from vertex $\mathbf{v}_k$ to the center of edge e_j. By normalizing the curvature integration with the weighted intersection lengths, we propose the normalized VNC-hard operator:

$$\hat{H}_{\text{hard}}(\mathbf{v}_k, r) = \frac{\sum_{j \in \mathcal{N}_r(k)} \alpha_j \cdot \delta_j \cdot w_{kj} \cdot \ell_j}{2 \sum_{j \in \mathcal{N}_r(k)} w_{kj} \cdot \ell_j} \tag{2}$$

Correspondingly, the average normal vector $\hat{\mathbf{n}}_{\text{hard}}(\mathbf{v}_k, r)$ is computed by identically weighting the area-based adjacent normals with $w_{kj} \cdot \ell_j$.



a) VNC-Scale curves, VNC-AUC, and the solution to VNC adaptive scales. b) Visualization of VNC adaptive scales, ensuring spatial balance.

Fig. 3: Computation of VNC-AUC and adaptive scales. The VNC-AUC metric $s(\mathbf{v}_k, \tau)$ accumulates multi-scale curvature information, while the adaptive scale $\tau_{adpt}(\mathbf{v}_k)$ indicates where the predefined information threshold is reached.

VNC-soft (Probabilistic Boundary & Matrix Tensorization). While VNC-hard provides exact bounded integration, computing explicit line-ball intersections ($\mathcal{N}_r(k)$ and ℓ_j) across multiple scales relies on intensive, non-differentiable spatial indexing (e.g., R-Trees). This irregular computational pattern hinders GPU parallelization on high-resolution meshes, creating a massive bottleneck for neural network integration.

To bridge this crucial gap between classical differential geometry and modern deep learning, we propose **VNC-soft**. Instead of defining a local neighborhood and explicitly computing partial edge lengths ℓ_j, VNC-soft computes the weighted sum over the *entire* edge set $\mathcal{E}$ of the mesh, utilizing the full topological edge lengths L_j:

$$\hat{H}_{\text{soft}}(\mathbf{v}_k, \sigma) = \frac{\sum_{j \in \mathcal{E}} \alpha_j \cdot \delta_j \cdot w_{kj}(\sigma) \cdot L_j}{2 \sum_{j \in \mathcal{E}} w_{kj}(\sigma) \cdot L_j} \tag{3}$$

Crucially, this mathematical relaxation drives a computational paradigm shift: *it transforms irregular, discrete geometric intersections into global, dense matrix multiplications.* By leveraging the Gaussian kernel's exponential decay to inherently ignore distant edges, explicit spatial queries become completely unnecessary. This enables VNC-soft to compute multi-scale curvature entirely via highly parallelizable tensor operations, aligning with deep learning frameworks.

To align VNC-soft with VNC-hard, we set a scaling relationship $r \approx \kappa\sigma$. Empirically, setting $\kappa = 6$ efficiently counteracts the quadratic growth of geometric elements at larger distances, ensuring an effective cut-off. A sensitivity analysis is provided in the supplement (Sec. A.4). Given its computational speedup and matrix-multiplication-friendly design while closely aligning with VNC-hard, VNC-soft serves as the practical engine for our intensive flow processing.

VNC for Multi-scale Geometric Analysis. To quantify geometric information across scales, we introduce two metrics as shown in Fig. 3a. First, the *VNC-Scale curve* $\mathcal{C}(\mathbf{v}_k)$ describes how perceived curvature $\hat{H}$ evolves with observation scale $\tau \in [\tau_{\min}, \tau_{\max}]$ (where τ refers to r in VNC-hard or $\kappa\sigma$ in

VNC-soft). Second, the *VNC-AUC metric* quantifies the local geometric complexity at a vertex, defined as the area between its VNC-Scale curve and the global mean curve $\overline{\hat{H}}(\tau)$:

$$s(\mathbf{v}_k, \tau) = \int_{\tau_{\min}}^{\tau} |\hat{H}(\mathbf{v}_k, x) - \overline{\hat{H}}(x)|\, dx \tag{4}$$

where the total complexity $\overline{s}(\mathbf{v}_k) \triangleq s(\mathbf{v}_k, \tau_{\max})$ serves as the governing indicator for our balanced flow algorithm in Sec. 3.2.

VNC Scale Stability. Grounded in integral geometry, our normalized VNC formulations yield a stable shape response across different neighborhood scales. By deliberately dividing the integrated curvature by a weighted length term, we inherently neutralize the scale magnitude, creating a *dimensionless* shape index $\hat{H}$. Consequently, the global average response $\overline{\hat{H}}_{\mathrm{VNC}}$ remains stable regardless of the scale. This average stability is precisely what matters in practice, as $\overline{\hat{H}}$ serves as a robust, non-drifting convergence target for neural shape simplification. The detailed analysis connecting this dimensionless property to classical integral invariants [26] is provided in the supplement (Sec. A.1).

3.2 Spacetime-Balanced Curvature Flow Sampling

The primary objective of the **VNC-Flow** algorithm is to tame the numerically irregular and topology-unstable curvature flow into structured, reliable deformation trajectories. This controlled sampling process serves a dual purpose: it establishes a physics-grounded prior for shape evolution, and provides predictable multi-scale intermediate states for the subsequent C-VAE to learn (Sec. 3.3).

Using the dimensionless VNC-soft operator $\hat{H}_{\mathrm{soft}}$ introduced in Sec. 3.1, we formulate the core vertex evolution step as:

$$\mathbf{v}_k \leftarrow \mathbf{v}_k - \lambda \cdot (\hat{H}_{\mathrm{soft}}(\mathbf{v}_k, \tau_k) - \overline{\hat{H}}) \cdot \hat{\mathbf{n}}_{\mathrm{soft}}(\mathbf{v}_k, \tau_k) \tag{5}$$

where $\overline{\hat{H}}$ is the scale-stable global reference, λ is the dynamic step size ensuring numerical stability, and τ_k is the assigned observation scale. This flow drives the surface toward global smoothness, minimizing $|\hat{H}_{\mathrm{soft}} - \overline{\hat{H}}| \to 0$. However, naively applying this flow with a uniform scale leads to unbalanced evolution rates, where flat regions may degenerate into singularities long before complex wrinkles are resolved. To overcome this, VNC-Flow operates under a *spacetime-information-balance* principle, structurally addressing three fundamental challenges:

(1) Spatial Balance (Adaptive Scale). To prevent regional over-smoothing, we enforce an *equal information-perception constraint.* Using the VNC-AUC metric to quantify local geometric information, we dynamically assign an adaptive scale $\tau_{\mathrm{adpt}}(\mathbf{v}_k)$ for each vertex in Eq. (6). Consequently, complex regions receive smaller scales to momentarily preserve fine structures, while flat regions receive larger scales to accelerate global smoothing.

$$\int_{\tau_{\min}}^{\tau_{\mathrm{adpt}}(\mathbf{v}_k)} |\hat{H}(\mathbf{v}_k, x) - \overline{\hat{H}}|\, dx = \varphi, \quad \forall k \in \{1, \ldots, N\} \tag{6}$$

Algorithm 1 Spacetime-Balanced Curvature Flow Sampling

Require: Input mesh $\mathcal{M}_{\text{gt}}$, scale range $[\tau_{\min}, \tau_{\max}]$, stages N_{stages}, step size λ
Ensure: Sequence of (Mesh, Normalized Time) pairs $\{(\mathcal{M}^{(i)}, t^{(i)})\}_{i=1}^{N_{\text{stages}}}$
Initialization:
  $\mathcal{M}^{(0)} \leftarrow \text{SDF-remesh}(\mathcal{M}_{\text{gt}})$ {Unify face count and clear topology}
  $\overline{S}_{\text{init}} \leftarrow \sum_{k=1}^{N} \overline{s}(\mathbf{v}_k \mid \mathcal{M}^{(0)})$
  $\varphi_{\text{stage}} \leftarrow \overline{S}_{\text{init}} / N_{\text{stages}}$ {Equidistant information decay budget}
for $i = 1$ to N_{stages} **do**
  Topology Regularization:
  $\mathcal{M}^{(i)} \leftarrow \text{SDF-remesh}(\mathcal{M}^{(i-1)})$ {Remesh if $i > 1$}
  Spatial Balance (Adaptive Scale):
  Compute $\tau_{\text{adpt}}(\mathbf{v}_k)$ for all $\mathbf{v}_k \in \mathcal{M}^{(i-1)}$ using Eq. (6)
  VNC-Flow Iteration:
  repeat
    $\mathbf{v}_k \leftarrow \mathbf{v}_k - \lambda \cdot (\hat{H}_{\text{soft}}(\mathbf{v}_k, \tau_{\text{adpt}}) - \overline{\hat{H}}) \cdot \hat{\mathbf{n}}_{\text{soft}}(\mathbf{v}_k, \tau_{\text{adpt}})$
    $\overline{S}_{\text{cur}} \leftarrow \sum_{k=1}^{N} \overline{s}(\mathbf{v}_k \mid \mathcal{M}^{(i)})$
  until $\overline{S}_{\text{init}} - \overline{S}_{\text{cur}} \geq \varphi_{\text{stage}} \cdot i$
  $t^{(i)} \leftarrow \overline{S}_{\text{cur}} / \overline{S}_{\text{init}}$ {Record normalized evolution time}
end for
return $\{(\mathcal{M}^{(i)}, t^{(i)})\}_{i=1}^{N_{\text{stages}}}$

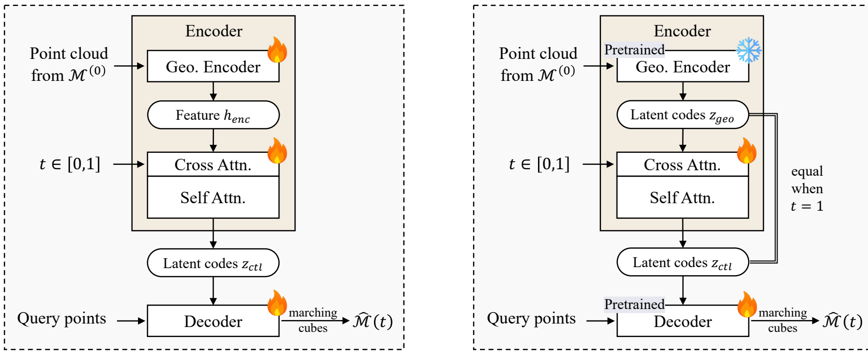
(2) Temporal Balance (Normalized Evolution Time). For a neural network to learn a continuous flow, the irregular physical time must be remapped into a predictable condition variable. We divide the flow into equidistant stages based on a geometric information decay budget. We define the *normalized evolution time* $t \in [0, 1]$ as the ratio of current geometric complexity to the initial complexity:

$$t^{(i)} = \frac{\sum_{k=1}^{N} \overline{s}(\mathbf{v}_k \mid \mathcal{M}^{(i)})}{\sum_{k=1}^{N} \overline{s}(\mathbf{v}_k \mid \mathcal{M}^{(0)})} = \frac{\overline{S}_{\text{cur}}^{(i)}}{\overline{S}_{\text{init}}} \tag{7}$$

where $t = 1$ represents the raw detailed mesh, and $t \to 0$ denotes the simplified base shape. This $t^{(i)}$ serves as the temporal coordinate to condition the C-VAE.

(3) Topology Regularization. Extreme mesh distortions during iteration inevitably cause numerical instability in discrete differential operators. We resolve this via *implicit topology regularization*, utilizing periodic SDF-based remeshing at stage transitions. This smoothly eliminates topological singularities while preserving the face count and surface area, thereby upholding the geometric prerequisites for VNC scale stability. It also regularizes triangle areas, mitigating VNC-soft/hard discrepancies on irregular tessellations.

Together, these three strategies are seamlessly unified in Algorithm 1. Spatial balance dictates *where* to smooth (via τ_{adpt}), temporal balance controls *when* to record a state (via $t^{(i)}$), and topology regularization ensures the mathematical integrity of the mesh. Consequently, VNC-Flow robustly bridges complex input geometry to its fundamental base shape, yielding high-quality, stage-wise continuous datasets critical for training the neural surrogate.



a) C-VAE-I: Intermediate Feature Fusion. b) C-VAE-II: Latent Code Modulation.

Fig. 4: Controllable VAE (C-VAE) architectures. Building upon the VecSet-based geometric backbone [13], our C-VAE introduces a temporal condition branch. **C-VAE-I** (Intermediate Feature Fusion) modulates encoder features via cross-attention with the temporal embedding t, tailored for dataset-level generalization. **C-VAE-II** (Latent Code Modulation) modulates independent high-dimensional latent codes utilizing a frozen pre-trained encoder, enabling high-fidelity trajectory distillation.

3.3 Controllable VAE for Structured Latent Space

To model the continuous geometric variations induced by the spacetime-balanced VNC-Flow, we propose a Controllable Variational Autoencoder (C-VAE) to act as the continuous neural surrogate. Instead of naive concatenation, our architecture employs a deep cross-attention mechanism governed by the normalized evolution time $t \in [0, 1]$ (Eq. (7)). To ensure versatility, we design two distinct conditioning pathways to fuse the temporal condition as shown in Fig. 4.

C-VAE-I: Intermediate Feature Fusion. Designed for robust dataset-level generation, this architecture injects the condition t into the encoder's intermediate feature layers via cross-attention. By modulating features early in the network, it efficiently learns global temporal relationships, making it highly suitable for extracting a generalized, compact latent space from diverse datasets.

C-VAE-II: Latent Code Modulation. Tailored for high-fidelity, subject-specific trajectory distillation, this variant leverages pre-trained geometric priors [13]. By freezing the backbone encoder, the cross-attention deeply modulates the extracted high-dimensional geometric latent codes $\mathbf{z}_{\text{geo}}$. This decoupled structure preserves the original generative capacity while enabling fine-grained, instance-level flow imitation without computationally expensive retraining.

Radial Latent Structure. Together with VNC-Flow supervision, this deep temporal modulation encourages the latent space to organize into a ***radial latent structure***. Unlike standard VAEs where arbitrary noise maps to unstructured shapes, our latent codes organize radially based on physical complexity. Latent codes naturally collapse toward the origin ($t \to 0$), representing minimum-information base topologies (e.g., spheres), while complex high-frequency details ($t \to 1$) anchor at the radial periphery. Consequently, a simple linear interpo-

Table 1: Comparison of curvature operators. VNC variants provide a practical balance of multi-scale capability, differentiability, and GPU-accelerated efficiency across representative mesh resolutions (10k, 25k, and 50k vertices).

Operator	Multi-scale	Parallel	Differentiable	V=10k	V=25k	V=50k
baseline [2]	✓	×	×	25 s	37 s	256 s
DiffCurvature [18]	×	✓	✓	**3 ms**	**4 ms**	**7 ms**
VNC-hard (Ours)	✓	✓	✓	149 ms	761 ms	3.0 s
VNC-soft (Ours)	✓	✓	✓	<u>57 ms</u>	<u>358 ms</u>	<u>1.4 s</u>

lation of t translates to a physically meaningful traversal along the non-linear geometric curvature flow.

Objective and Training Strategy. Following standard practices [13, 34], the C-VAE is optimized using a combined loss $\mathcal{L} = \mathcal{L}_{\text{tsdf}} + \beta_{\text{kl}}\mathcal{L}_{\text{kl}} + \lambda_{\text{eik}}\mathcal{L}_{\text{eik}}$. For C-VAE-I, we employ a temporal curriculum learning strategy on β_{kl} to gently enforce the formation of the radial latent structure. For C-VAE-II, we utilize a distillation paradigm, optimizing purely the attention modules and decoder while freezing the encoder. Detailed network configurations, constraints, and loss derivations are provided in the supplement (Sec. A.3).

4 Experiments

4.1 Evaluation of the VNC Operator

We compare our VNC operators against a classical CPU-based integral mean curvature [2] (Baseline) and a recent differentiable curvature-based operator [18] (DiffCurvature). Timings are measured on an RTX 3090 GPU and an Intel Xeon Gold CPU using processed THuman2.0 meshes [33].

Computational Efficiency. As shown in Tab. 1, while DiffCurvature is exceptionally fast, it lacks multi-scale support. Conversely, the classical baseline provides multi-scale curvature estimation but becomes computationally prohibitive as mesh resolution increases. Our VNC variants bridge this gap by jointly offering differentiability, GPU parallelism, and explicit multi-scale capability. In particular, the matrix-tensorized VNC-soft consistently achieves a 2–3× speedup over VNC-hard, which itself is already 50–150× faster than the CPU baseline. This makes VNC-soft well suited for intensive flow sampling.

Multi-scale Decoupling & Scale Stability. Fig. 5 visually confirms that VNC acts as an effective geometric frequency filter, naturally transitioning from capturing high-frequency wrinkles at small scales to low-frequency structural topology at large scales. Crucially, Fig. 6 validates our theoretical claim: VNC functions as a *dimensionless shape index*. Unlike the baseline, the global mean of our normalized VNC ($\overline{\hat{H}}$) remains impressively stable across varying scales and diverse topological inputs such as SMPL-X [25] body and clothed humans [33]. This average scale stability provides a highly reliable, non-drifting convergence target for the flow evolution.

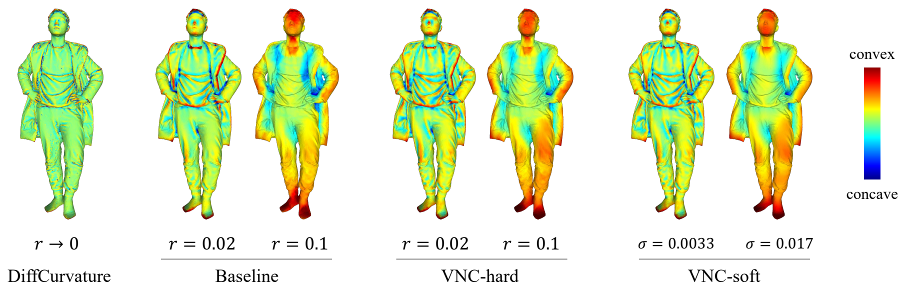
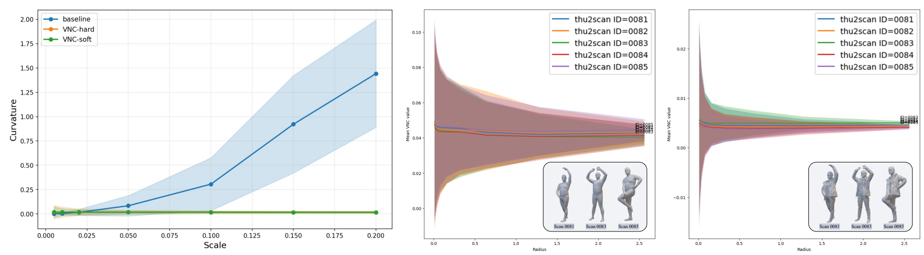


Fig. 5: Curvature operators across scales. VNC transitions from high-frequency wrinkles at small scales to low-frequency abstractions at large scales. Setting $r = 6\sigma$ effectively aligns VNC-soft with explicitly bounded VNC-hard.



a) Scale-space analysis of curvature mean and variance: baseline vs. VNC-hard vs. VNC-soft.

b) Scale-space analysis of VNC mean and variance for SMPL-X meshes.

c) Scale-space analysis of VNC mean and variance for clothed human meshes.

Fig. 6: Validation of Dimensionless Scale Stability. (a) Unlike the baseline whose mean curvature diverges with scale, VNC operators maintain a stable global mean while variance strictly decreases. (b, c) This scale-stable property holds robustly across diverse topological inputs, ensuring a reliable convergence target.

4.2 Evaluation of Spacetime-Balanced VNC-Flow

We compare VNC-Flow against fixed-scale variants and mean-curvature-flow baselines, then analyze how adaptive scales together with the stage-wise information budget produce spacetime-balanced trajectories.

Comparisons and Ablations. Fig. 7 compares adaptive-scale VNC-Flow with fixed-scale variants and two mean-curvature-flow baselines on a complex human shape. The fixed-scale variants fail to reach the spherical target, exhibiting uneven degeneration at a small scale (c) or incomplete abstraction at a large scale (d). The mean-curvature-flow baselines either collapse (e) or contract toward a mean-curvature skeleton (f). In contrast, adaptive-scale VNC-Flow progressively approaches the target while maintaining valid intermediate surfaces (b). These results support the combined use of the scale-stable target $\overline{\hat{H}}$ and spatially adaptive scales.

Spacetime Balance and Continuous Abstraction. Fig. 8a shows an overall reduction in mean VNC-AUC, with stage transitions defined by the

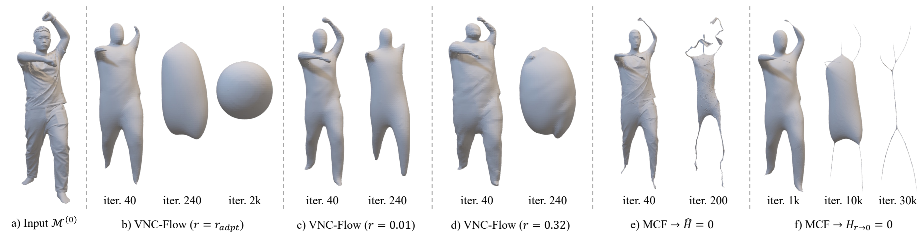


Fig. 7: Comparison and ablation of curvature-flow variants. (a) Input shape. (b) Adaptive-scale VNC-Flow approaches the spherical target. Fixed-scale variants exhibit (c) uneven degeneration or (d) incomplete abstraction, while mean-curvature-flow baselines (e) collapse or (f) contract toward a mean-curvature skeleton.

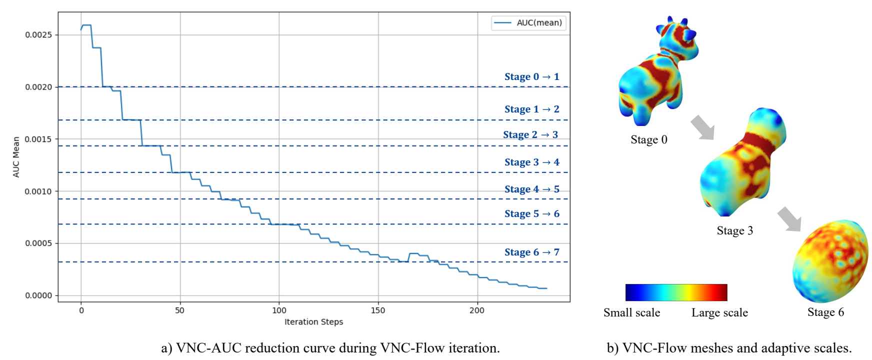


Fig. 8: Spacetime balance in VNC-Flow. (a) Mean VNC-AUC decreases overall across the stage thresholds. Small rebounds arise at SDF-based remeshing boundaries. (b) Adaptive-scale maps assign smaller scales to geometrically complex regions and larger scales to smoother regions.

information-decay budget. The small rebounds near some transitions arise from SDF-based remeshing and do not alter the global trend. Spatially, Fig. 8b shows that complex regions receive smaller scales while smoother regions receive larger scales. Across human scans with diverse poses and clothing, Fig. 9a further shows that this ordered evolution consistently produces continuous abstraction trajectories toward compact base shapes, providing structured intermediate states for training the neural surrogate.

4.3 C-VAE Trajectory Learning and Latent Space Anatomy

To comprehensively evaluate the efficacy of our temporal-conditioned latent mapping, we assess the C-VAE across two distinct scenarios: dataset-level generalization and subject-specific high-fidelity trajectory distillation.

Generalization Capability (C-VAE-I). We first evaluate the cross-subject generalizability of C-VAE-I, configured with $\mathbf{z}_{\text{ctl}}^{(1)} \in \mathbb{R}^{2048\times 32}$. This model is

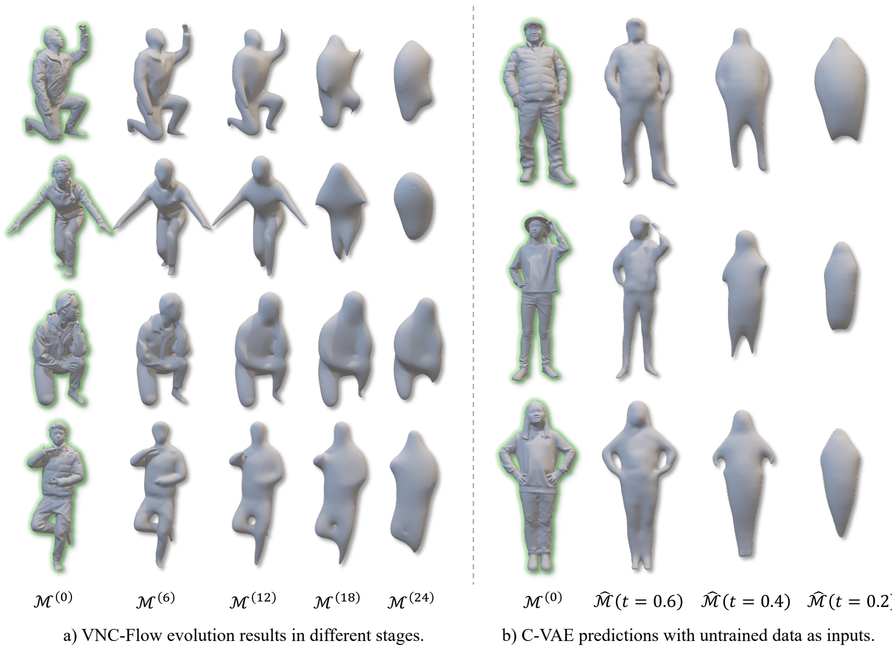


Fig. 9: VNC-Flow evolution and C-VAE predictions. (a) VNC-Flow produces ordered abstraction trajectories for human scans with diverse poses and clothing. (b) On unseen scans, the trained C-VAE-I predicts continuous forward abstraction trajectories conditioned solely on the normalized time t.

trained from scratch on 32-stage VNC-Flow sequences generated from 500 THuman2.0 scans [33] (normalized to 50k faces to satisfy VNC prerequisites). We then evaluate this model on the entirely unseen 2K2K dataset [9], featuring different poses and clothing topologies. As shown in Fig. 9b, by modulating the normalized time condition $t \in [0, 1]$, the network reliably predicts the continuous forward abstraction process, showing it has internalized the physical prior of the curvature flow beyond memorizing the training set.

High-Fidelity Subject-Specific Trajectory (C-VAE-II). To evaluate the representational capacity of our temporal mapping, we utilize C-VAE-II to intentionally overfit a highly complex sequence, configuring $\mathbf{z}_{\text{ctl}}^{(2)} \in \mathbb{R}^{2048\times 64}$ to match the latent space dimensionality of the pretrained weights [13]. Fig. 10 qualitatively compares our paradigm against NMF [8], NIE [19], and MeshSDF [27]. While existing methods suffer from fixed topologies, non-geometric artifacts, or unphysical smoothing orders, our C-VAE closely follows the physical curvature flow. Quantitatively, Tab. 2 demonstrates that our temporal conditioning ($t = 1$) yields significantly higher reconstruction precision (CD, EMD, and NCE [6, 20]) compared to baseline discrete iterations or latent interpolations.

Table 2: Quantitative comparison of surface evolution methods. Our C-VAE-II achieves superior reconstruction accuracy at the target shape ($t = 1$).

Method	CD$\times 10^{-4}$ ($\downarrow$)	EMD$\times 10^{-2}$ ($\downarrow$)	NCE (rad) ($\downarrow$)
NMF [8] (iter. 15k)	1.92	5.60	0.518
NIE [19] (iter. 1k)	4.77	3.17	0.405
MeshSDF [27] ($\alpha = 1.0$)	1.61	2.55	0.341
C-VAE-II (Ours, $t = 1.0$)	**1.05**	**1.94**	**0.257**

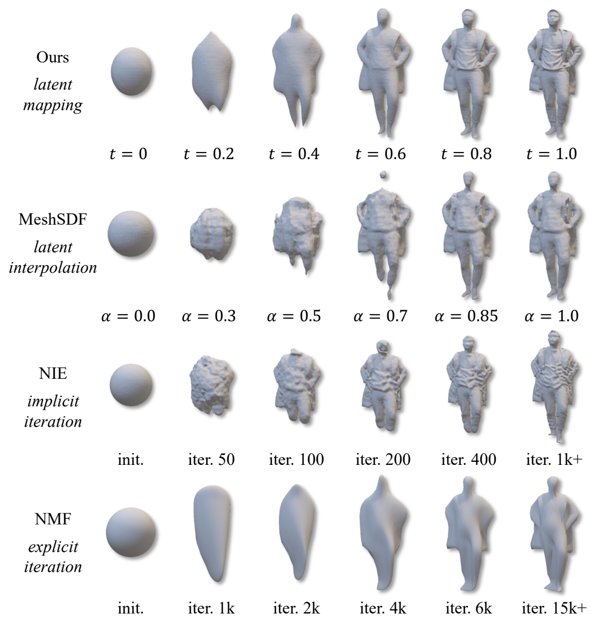
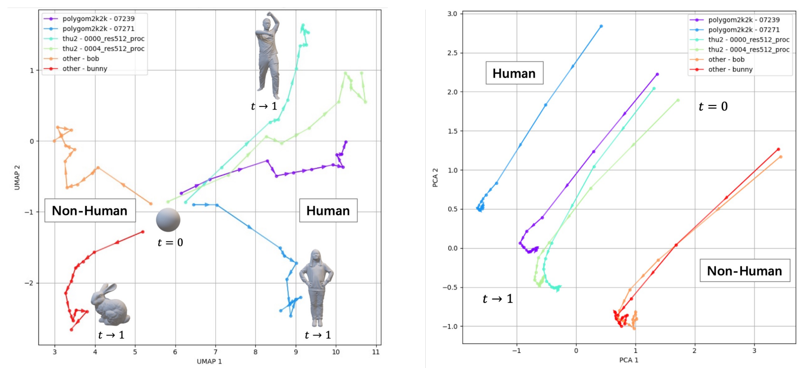


Fig. 10: Comparison of trajectory-learning paradigms. Compared to existing paradigms, our C-VAE follows the physically grounded, frequency-ordered decay.

Anatomy of the Radial Latent Space. To examine the organization of the learned latent space $\mathbf{z}_{\mathrm{ctl}}^{(1)} \in \mathbb{R}^{2048\times 32}$, we apply two feature aggregation strategies followed by dimensionality reduction. Max-pooling with UMAP (Fig. 11a) reveals a *radial organization*: minimum-information base shapes at $t \to 0$ cluster near the center, while increasing geometric detail extends toward the periphery. Mean-pooling with PCA (Fig. 11b) further reveals a shared global direction associated with t. This organization is primarily enabled by VNC-Flow, which aligns subjects through the normalized evolution coordinate and a common base-shape endpoint. Together with temporal conditioning, this alignment provides consistent cross-subject supervision and reduces the subject-specific variation retained at $t \to 0$. These observations suggest that shape identity and evolutionary stage are structurally organized in the learned latent space.



a) UMAP visualization of C-VAE latent space. b) PCA visualization of C-VAE latent space.

Fig. 11: Latent-space anatomy of C-VAE-I. (a) UMAP reveals a *radial structure*, anchoring base topologies at the center and complex features at the periphery. (b) PCA reveals a *parallel structure*, indicating a unified latent direction for the time condition.

5 Discussion and Conclusion

Several aspects of our framework merit further discussion. **(1) Scope and Limitations:** The SDF-based topology regularization assumes watertight inputs, therefore fragmented raw scans require a robust implicit conversion before evolution. Moreover, because our experiments focus on human scans, C-VAE-I demonstrates within-domain generalization across poses and clothing, while its transfer across object categories remains to be established. **(2) Applications and Future Directions:** The ordered progression from minimum-information base shapes ($t \to 0$) to detailed surfaces ($t \to 1$) naturally supports level-of-detail control, geometric abstraction, and frequency-aware editing. While our current pipeline learns from offline VNC-Flow trajectories, the differentiable VNC-soft operator could support future gradient-based objectives. Combining this scale-aware prior with generative ODEs such as Flow Matching [1, 16] may further enable controllable 3D generation.

In conclusion, we introduce a learnable scale-space framework that connects classical differential geometry with neural 3D representation. The parallelizable VNC operator and spacetime-balanced VNC-Flow convert irregular physical smoothing into structured training trajectories, which the C-VAE learns as a time-conditioned continuous surrogate. Together, these components provide an interpretable basis for controllable, topology-stable shape abstraction and a scale-aware prior for future 3D generative models.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (NSFC) under Grant 62176156.

References

1. Chen, R.T.Q., Lipman, Y.: Flow matching on general geometries. In: International Conference on Learning Representations (2024)
2. Cohen-Steiner, D., Morvan, J.m.: Restricted delaunay triangulations and normal cycle. In: Proceedings of Symposium on Computational Geometry. pp. 312–321 (2003)
3. Colding, T., Minicozzi, W., Pedersen, E.: Mean curvature flow. Bulletin of the American Mathematical Society **52**(2), 297–333 (2015)
4. Dong, Z., Chen, X., Yang, J., Black, M.J., Hilliges, O., Geiger, A.: AG3D: Learning to generate 3d avatars from 2d image collections. In: Proceedings of IEEE International Conference on Computer Vision. pp. 14870–14881 (2023)
5. Duarte, M.E., Sacht, L.: Mean curvature flow and applications. Universidade Federal de Santa Catarina, Departamento de Matemática, Florianópolis, Brazil (2017)
6. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. pp. 605–613 (2017)
7. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. In: Proceedings of International Conference on Machine Learning. pp. 3789–3799 (2020)
8. Gupta, K., Chandraker, M.: Neural Mesh Flow: 3d manifold mesh generation via diffeomorphic flows. In: Advances in Neural Information Processing Systems. pp. 1747–1758 (2020)
9. Han, S.H., Park, M.G., Yoon, J.H., Kang, J.M., Park, Y.J., Jeon, H.G.: High-fidelity 3d human digitization from single 2k resolution images. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. pp. 12869–12879 (2023)
10. Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., Tan, H.: LRM: Large reconstruction model for single image to 3d. In: International Conference on Learning Representations (2024)
11. Kazhdan, M., Solomon, J., Ben-Chen, M.: Can mean-curvature flow be modified to be non-singular? Computer Graphics Forum **31**(5), 1745–1754 (2012)
12. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
13. Li, Y., Zou, Z.X., Liu, Z., Wang, D., Liang, Y., Yu, Z., Liu, X., Guo, Y.C., Liang, D., Ouyang, W., Cao, Y.P.: TripoSG: High-fidelity 3d shape synthesis using large-scale rectified flow models. arXiv preprint arXiv:2502.06608 (2025)
14. Liao, T., Yi, H., Xiu, Y., Tang, J., Huang, Y., Thies, J., Black, M.J.: TADA! text to animatable digital avatars. In: Proceedings of IEEE International Conference on 3D Vision (2024)
15. Lindeberg, T.: Scale-space theory: A basic tool for analyzing structures at different scales. Journal of Applied Statistics **21**(1-2), 225–270 (1994)
16. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: International Conference on Learning Representations (2023)
17. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. ACM SIGGRAPH Computer Graphics **21**(4), 163–169 (1987)
18. Luo, Y.: Differentiable topology estimating from curvatures for 3d shapes. arXiv preprint arXiv:2412.00140 (2024)

19. Mehta, I., Chandraker, M., Ramamoorthi, R.: A level set theory for neural implicit evolution under explicit flows. In: Proceedings of European Conference on Computer Vision. pp. 711–729 (2022)
20. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. pp. 4455–4465 (2019)
21. Meyer, M., Desbrun, M., Schröder, P., Barr, A.H.: Discrete differential-geometry operators for triangulated 2-manifolds. In: Visualization and Mathematics III. pp. 35–57 (2003)
22. Novello, T., Silva, V., Schardong, G., Schirmer, L., Lopes, H., Velho, L.: Neural implicit surface evolution. In: Proceedings of IEEE International Conference on Computer Vision. pp. 14279–14289 (2023)
23. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. Journal of Computational Physics **79**(1), 12–49 (1988)
24. Paschalidou, D., Katharopoulos, A., Geiger, A., Fidler, S.: Neural Parts: Learning expressive 3d shape abstractions with invertible neural networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. pp. 3204–3215 (2021)
25. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. pp. 10967–10977 (2019)
26. Pottmann, H., Wallner, J., Yang, Y.L., Lai, Y.K., Hu, S.M.: Principal curvatures from the integral invariant viewpoint. Computer Aided Geometric Design **24**(8-9), 428–442 (2007)
27. Remelli, E., Lukoianov, A., Richter, S., Guillard, B., Bagautdinov, T., Baque, P., Fua, P.: MeshSDF: Differentiable iso-surface extraction. In: Advances in Neural Information Processing Systems. pp. 22468–22478 (2020)
28. Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.: Implicit neural representations with periodic activation functions. In: Advances in Neural Information Processing Systems. pp. 7462–7473 (2020)
29. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: LGM: Large multi-view gaussian model for high-resolution 3d content creation. In: Proceedings of European Conference on Computer Vision. pp. 1–18. Springer (2024)
30. Wu, K., Liu, F., Cai, Z., Yan, R., Wang, H., Hu, Y., Duan, Y., Ma, K.: Unique3D: High-quality and efficient 3d mesh generation from a single image. In: Advances in Neural Information Processing Systems. vol. 37, pp. 125116–125141 (2024)
31. Xu, J., Cheng, W., Gao, Y., Wang, X., Gao, S., Shan, Y.: InstantMesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. arXiv preprint arXiv:2404.07191 (2024)
32. Yang, G., Belongie, S., Hariharan, B., Koltun, V.: Geometry processing with neural fields. In: Advances in Neural Information Processing Systems. pp. 22483–22497 (2021)
33. Yu, T., Zheng, Z., Guo, K., Liu, P., Dai, Q., Liu, Y.: Function4D: Real-time human volumetric capture from very sparse consumer rgbd sensors. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. pp. 5742–5752 (2021)
34. Zhang, B., Tang, J., Niessner, M., Wonka, P.: 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. ACM Transactions on Graphics **42**(4), 1–16 (2023)