Motion2VecSets: Non-Rigid Shape Reconstruction and Tracking With 4D Latent Set Diffusion

Jiapeng Tang, Wei Cao, Biao Zhang, Chang Luo, Yaoyao Liu, *Member, IEEE*, and Matthias Nießner

***Abstract*—We introduce Motion2VecSets, a 4D diffusion model for dynamic surface mesh generation from various ambiguous observations, including a sequence of RGB images, sparse and partial point clouds, and low-resolution voxel grids. While recent methods using neural field representations have shown success in modeling non-rigid objects, conventional feed-forward architectures struggle with noisy, partial, or sparse observations due to their deterministic nature. To address the inherent one-to-many mapping problem, we introduce a diffusion model that explicitly learns the shape and motion distribution of non-rigid objects through an iterative denoising process of compressed latent representations. The diffusion-based priors provide more plausible and diverse reconstructions under ambiguous conditions. Instead of relying on global latent codes, we represent 4D dynamics using latent sets. This novel 4D representation captures local shape and deformation patterns, leading to more accurate non-linear motion capture and significantly improving generalization capacity to unseen motions and identities. For temporally coherent tracking, we jointly denoise latent sets across frames and enable cross-frame information exchange. To reduce computational cost, we design an interleaved spatial-temporal attention block that alternately aggregates deformation latents along spatial and temporal dimensions. Extensive experiments on datasets of humans, animals, and articulated objects demonstrate that Motion2VecSets outperforms prior methods in reconstructing and tracking non-rigid deformations from various imperfect observations.**

***Index Terms*—Diffusion models, non-rigid object reconstruction and tracking, 3D and 4D surface generation, mesh deformation.**

Received 25 August 2025; revised 29 January 2026; accepted 28 March 2026. Date of publication 6 April 2026; date of current version 6 July 2026. The work of Wei Cao and Yaoyao Liu was supported in part by the National Artificial Intelligence Research Resource (NAIRR) Pilot under Award NAIRR250199 and in part by Delta and DeltaAI at the National Center for Supercomputing Applications (NCSA) through ACCESS allocations under Award CIS250012, Award CIS250816, and Award CIS251188. This work was supported by ERC Consolidator Grant Gen3D under Grant 101171131. Recommended for acceptance by J. Gu. *(Jiapeng Tang, Wei Cao, Biao Zhang, and Chang Luo contributed equally to this work.) (Corresponding authors: Jiapeng Tang; Wei Cao.)*

Jiapeng Tang, Chang Luo, and Matthias Nießner are with the Technical University of Munich, 80539 Munchen, Germany (e-mail: jiapeng.tang@tum.de; chang.luo@tum.de; niessner@tum.de).

Wei Cao and Yaoyao Liu are with the University of Illinois Urbana-Champaign, Champaign, IL 61820 USA (e-mail: weicao3@illinois.edu; lyy@illinois.edu).

Biao Zhang is with the King Abdullah University of Science and Technology, Thuwal 23955, Saudi Arabia (e-mail: biao.zhang@kaust.edu.sa).

Our implementation is available at https://vveicao.github.io/projects/Motion2VecSets.

This article has supplementary downloadable material available at https://doi.org/10.1109/TPAMI.2026.3680779, provided by the authors.

Digital Object Identifier 10.1109/TPAMI.2026.3680779

I. Introduction

RECONSTRUCTING dynamic object surfaces and motions from diverse observations, such as point clouds, voxel grids, and images, is a core research area of computer vision and computer graphics. It plays a vital role in practical applications like virtual and augmented reality, computer games, movie effects, and robotic manipulation.

Classic methods for 3D mesh reconstruction and generation trace back to foundational techniques such as Marching Cubes [1], Poisson Surface Reconstruction [2], KinectFusion [3], and Multi-view Stereo [4]. For non-rigid object reconstruction and tracking, approaches like DynamicFusion [5], VolumeDeform [6], and DoubleFusion [7] jointly perform surface fusion and motion tracking by leveraging handcrafted deformation priors, such as As-Rigid-As-Possible (ARAP) [8] and Embedded Deformation [9]. While these methods are effective for capturing short-term motions, they often struggle to handle complex or highly non-linear deformations commonly encountered in real-world scenarios.

Recently, there have been notable advances in learning-based 3D and 4D object modeling. Early efforts employed parametric models [10], [11], [12], [13], [14] tailored to specific object categories. However, their reliance on a fixed mesh topology limits their ability to capture the complex 4D dynamics of general non-rigid objects. Model-free methods [15], [16], [17] overcome this constraint by using coordinate-based MLPs [18] to model deformations with an arbitrary topology and an unstructured geometry, showing promising results for large non-rigid motions. Despite these advances, current methods still face challenges under ambiguous input conditions, such as noisy, sparse, or partial observations, where the reconstruction problem becomes ill-posed due to multiple plausible solutions. Moreover, they represent dynamics as a sequence of single latent codes and thus struggle to capture shape and motion priors accurately. These issues become even more severe with unseen identities, due to the limited generalization capacity of the global latent representation.

To address the aforementioned challenges, we propose Motion2VecSets, a 4D diffusion model designed for dynamic surface mesh reconstruction from various ambiguous observations, including sparse, noisy, or partial point clouds, RGB images, and coarse voxel grids. It explicitly learns a probabilistic distribution of non-rigid surface geometry and temporal dynamics via an iterative denoising process, enabling more realistic and diverse reconstructions, especially in the presence of uncertain

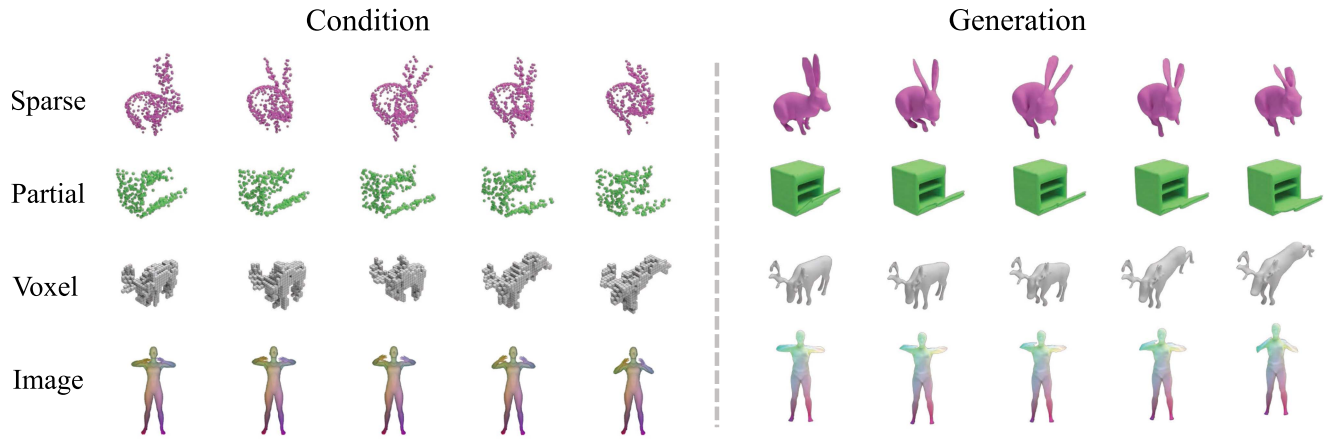


Fig. 1. We present *Motion2VecSets*, a 4D diffusion model capable of reconstructing dynamic surface meshes with complex geometries and robust motion tracking from ambiguous 4D observations, including sequences of sparse and partial point clouds, coarse voxel grids, and RGB images.

inputs. Learning such a 4D diffusion requires a compact yet expressive representation to encode the underlying shape and motion. Inspired by the observation that objects with diverse topologies often exhibit similar local geometry and deformation patterns, we represent dynamic surfaces as a sequence of latent sets: one for capturing the geometry of the initial frame and the others for modeling its temporal evolution. This design not only preserves local shape and motion details but also improves the generalization capacity to unseen identities and motions. We employ a shape latent set diffusion model to reconstruct the 3D surface mesh at the initial frame. For non-rigid deformation tracking, we introduce a synchronized deformation latent set diffusion, which jointly denoises deformation latent sets across all time frames. This enforces spatio-temporal consistency and enables coherent surface tracking throughout the sequence. To address the high memory cost associated with simultaneous deformation diffusion across time, we propose an interleaved space-time attention block as the core unit of the denoiser. This module alternates between aggregating latent features along the spatial and temporal dimensions, achieving efficient yet expressive modeling. As illustrated in Fig. 1, our Motion2VecSets can reconstruct plausible, high-fidelity non-rigid surfaces with complex structures and exhibits robust motion tracking performance under a variety of ambiguous input settings.

A preliminary version of this work was presented at CVPR 2024, where we introduced Motion2VecSets [24] for dynamic shape reconstruction from point cloud sequences. In this extended version, we expand both the method and experimental scope. Motion2VecSets now supports 4D mesh reconstruction from a wide range of imperfect 4D observations, including noisy, sparse, or partial point clouds, coarse voxel grids, and RGB images. A comparison of 3D and 4D reconstruction methods under different input settings is provided in Table I. We expand our experiments to show that our Motion2VecSets can handle a broader range of non-rigid objects, including humanoids, animals, and articulated objects. We further broaden our evaluation to demonstrate the generality of our framework across diverse non-rigid object categories, including humans, animals, and articulated objects. Additionally, we show that Motion2VecSets can synthesize natural surface motions when conditioned on sparse 3D handle trajectories. Finally, we conduct robustness experiments under varying levels of noise and sparsity in point cloud inputs, demonstrating that our method consistently outperforms state-of-the-art approaches. Our contributions can be summarized as follows:

TABLE I
COMPARISON OF DIFFERENT 3D AND 4D RECONSTRUCTION METHODS

Method	#Latents	Model-Free	4D Recon.	Inputs			Probabilistic Outputs
				Points	Images	Voxels	
SMPL [10]	–	✗	✓	–	–	–	–
MANO [12]	–	✗	✓	–	–	–	–
DeepSDF [19]	Single	✓	✗	✓	✗	✗	✗
OccNet [20]	Single	✓	✗	✓	✓	✓	✓
ConvOccNet [21]	Multiple	✓	✗	✓	✗	✓	✓
3DShape2VecSet [22]	Multiple	✓	✗	✓	✓	✗	✓
Oflow [15]	Single	✓	✓	✓	✓	✗	✓
LPDC [16]	Single	✓	✓	✓	✗	✗	✗
CaDex [17]	Single	✓	✓	✓	✗	✗	✗
DNF [23]	Single	✓	✓	✗	✗	✗	✗
Ours	Multiple	✓	✓	✓	✓	✓	✓

- We propose a novel **4D latent diffusion model** for dynamic surface mesh generation, enabling realistic and consistent shape deformation over time.
- We introduce a novel **4D neural representation based on latent sets**, coupled with transformer architectures. This representation improves the capacity to model complex geometry and motion, and enhances generalization to unseen identities and dynamics.
- We design an **Interleaved Spatio-Temporal Attention** mechanism to denoise multi-frame bundled deformation latent sets. This ensures coherent spatio-temporal structure across frames while maintaining high computational efficiency.
- We demonstrate the effectiveness of our method in **non-rigid shape reconstruction and tracking** from diverse and imperfect 4D observations, including a sequence of sparse and partial point clouds, RGB images, and low-resolution voxel grids.

Extensive comparisons with state-of-the-art methods show that Motion2VecSets achieves superior performance in dynamic

surface reconstruction across various object categories, including humanoids, animals, and articulated shapes. It outperforms prior methods on commonly used benchmarks such as Dynamic FAUST [25], DeformingThings4D-Animals [26], and Shape2Motion [27].

II. Related Works

In this section, we review these closely related works, including 3D shape representation and reconstruction, non-rigid deformation and tracking, diffusion models, and 3D/4D shape generation.

A. 3D Shape Representation and Reconstruction

Over the past decade, learning-based 3D reconstruction methods have explored a variety of shape representations, including voxels [28], [29], [30], [31], point clouds [32], [33], [34], octrees [35], [36], [37], and meshes [38], [39], [40], [41], [42], [43]. Meshes are the most natural and compact representation of 3D surfaces and are widely used in downstream applications such as rendering, simulation, and animation. However, learning to directly predict high-quality mesh topology and geometry remains challenging due to their irregular structure and sensitivity to discretization artifacts. Alongside general-purpose 3D reconstruction methods, parametric models have proven effective for modeling specific shape categories, such as the human body (e.g., SMPL [10], STAR [11]), face (e.g., FLAME [13]), hand (e.g., MANO [12]), and animal (e.g., SMAL [14]). These models offer strong performance for tasks like tracking and animation within their predefined domains. However, their reliance on fixed mesh templates limits their ability to capture large topological variations and complex non-rigid deformations observed in general object categories. To address these limitations, recent advances in neural implicit function fields [18], [44], [45] have gained widespread attention. These approaches represent 3D geometry as continuous fields using coordinate-based MLPs, such as occupancy fields [18], signed distance functions (SDFs) [45], and implicit function decoders [44]. Such representations provide high flexibility in modeling arbitrary topologies and fine-grained geometric details, and can reconstruct surfaces at theoretically infinite resolution. Subsequent works have extended these ideas with improved network architectures and training strategies [46], [47], [48], [49], [50], [51], further pushing the boundaries of high-fidelity 3D reconstruction.

B. Non-Rigid Deformation and Tracking

Early methods for non-rigid surface tracking, such as DynamicFusion [5], DoubleFusion [7], and VolumeDeform [6], jointly perform surface fusion and motion tracking from RGB-D sequences. These approaches rely on predefined deformation priors, typically assuming local rigidity to regularize motion. Common regularization strategies include As-Rigid-As-Possible (ARAP) [8] and Embedded Deformation (ED) [9], which enable real-time performance and smooth deformation tracking under limited motion. However, such hand-crafted priors struggle to generalize to complex, large-scale, or non-linear deformations commonly found in real-world scenarios. To address these limitations, recent works have explored data-driven approaches. Notably, Neural Shape Deformation Priors (NSDP) [52] introduce learned deformation priors from training data, demonstrating improved capability in capturing highly non-linear and topology-aware deformations. Recent advancements in 3D shape representation have been effectively extended to the 4D domain, enabling a more comprehensive modeling of object dynamics over time, such as OFlow [15], NPMs [53], LCRODE [54], LPDC [16], CaDeX [17]. Despite their success, these methods rely on a single global latent code [16], [17], [53] to encode temporal shape variations, which limits their ability to capture fine-grained local deformations and generalize across diverse identities or motion patterns. In contrast, we represent recurring local geometric and dynamic patterns across different objects using a set of latent codes. Our method captures complex surfaces and motions with higher accuracy and generalizes well to unseen identities and motions.

C. Diffusion Models

Diffusion models [55], [56] have recently emerged as a prominent class of generative frameworks based on iterative denoising processes. They have achieved state-of-the-art results across diverse domains, including image synthesis [57], [58], [59], [60], [61], video generation [62], [63], [64], audio generation [65], [66], and text creation [67], [68], [69]. To improve scalability and reduce the computational cost associated with high-resolution synthesis, Latent Diffusion Models (LDMs) [60] encode data into compact latent spaces, enabling efficient high-resolution synthesis. For video generation, early adaptations like Video Diffusion Models [64] and Imagen Video [70] apply denoising independently per frame, often resulting in temporal inconsistency and motion artifacts, while also incurring high costs for long sequences. To address this, recent approaches introduce motion-aware mechanisms: Stable Video Diffusion [62] incorporates temporal attention in latent space, and AnimateDiff [63] leverages pre-trained motion modules to animate diffusion outputs coherently. Our synchronized deformation diffusion shares a similar motivation with recent video diffusion models. We introduce an interleaved attention mechanism that alternates between spatial and temporal dimensions, substantially reducing computational complexity and memory consumption while preserving temporal coherence.

D. 3D and 4D Generation

Recent advancements in generative models have been extended to 3D and 4D content generation, demonstrating strong performance across a variety of tasks, including shape generation [22], [71], [72], [73], scene synthesis [74], texture and material generation [75], [76], [77], human generation [78], [79], [80], and human scene interaction [81], [82], [83], [84].

Early efforts [71], [72], [85] applied the diffusion process directly to point cloud denoising of a fixed size. Other approaches optimize 3D representations, such as NeRF [86] or 3D Gaussian

Splatting [87], by leveraging pre-trained text-to-image foundation models [62] via Score Distillation Sampling (SDS) [88] and its variants. To enable feed-forward mesh generation without costly per-shape optimization, subsequent methods [22], [73], [89] compress 3D shapes into latent representations, which are then decoded into neural fields, where the denoising process is performed in latent space. LaGeM [90] and Structured 3D Latents [91] further organize latent codes hierarchically to capture structural information at multiple semantic levels. Recent large-scale models such as Direct3D [92] and Hunyuan3D [93], [94] scale latent diffusion to high-resolution textured asset generation, using transformer-based architectures and multimodal training on massive 3D datasets.

For non-rigid object generation, NAP [95] formulates it as a graph generation task, including both nodes and edges. Similarly, methods such as RigAnything [96] and MagicArticulate [97] enable explicit skeleton and skinning weight prediction from raw meshes. However, they still primarily focus on static generation, without modeling temporal consistency across deformations. A separate line of work targets human motion generation [23], [98], [99], [100]. In contrast, our focus lies in modeling detailed surface deformations rather than highly abstracted skeletal motions. Additionally, our method addresses general non-rigid objects beyond the human domain. Most relevant to our work, DNF [23] disentangles geometry and motion using a dictionary-based shared latent field and learns unconditional 4D object generation. However, unlike DNF, our approach focuses on conditional 4D reconstruction and tracking from diverse and ambiguous inputs such as sparse point clouds, coarse voxels, and RGB images. Other works, such as L4GM [101], Diffusion4D [102], and Consistent4D [103], focus on optimizing NeRF or 3D Gaussian Splatting (3DGS) representations from monocular videos. In contrast, our work emphasizes high-quality surface geometry reconstruction and robust tracking, rather than volumetric radiance modeling or image-based view synthesis.

III. Approach

The objective is to generate surface meshes with dense temporal correspondences from various imperfect inputs, including coarse voxel grids, noisy or partial point clouds, and RGB images. The reconstructed mesh sequence is denoted as $\{\mathcal{M}^t\}_{t=1}^T = \{\mathcal{V}^t, \mathcal{F}^t\}_{t=1}^T$, where $\mathcal{V}^t$ and $\mathcal{F}^t$ represent the vertices and faces at each time step t. We aim to address this problem via neural fields. An occupancy field and a series of flow fields parameterize an object mesh sequence:

$$\begin{aligned} \text{Occupancy}: &\quad \mathbb{R}^3 \to \mathbb{R}, \\ \text{Flow}_{1\to 2}: &\quad \mathbb{R}^3 \to \mathbb{R}^3, \\ &\vdots \\ \text{Flow}_{1\to T}: &\quad \mathbb{R}^3 \to \mathbb{R}^3. \end{aligned} \tag{1}$$

The occupancy field describes the implicit surface of the first frame $\mathcal{M}^1 = \{\mathcal{V}^1, \mathcal{F}^1\}$, which can be easily extracted using iso-surface extraction algorithms. The flow fields are densely defined on the surface of the initial frame and output a vector on the target shape. Subsequent mesh frames can be recovered by querying the flow field. The symbol $\text{Flow}_{1\to t}$ gives the flow field between frame 1 and frame t. They are used to deform $\mathcal{V}^1$ to $\{\mathcal{V}\}_{t=2}^T$. The face connectivity $\mathcal{F}^t$ is inherited from $\mathcal{F}^1$.

Conventional feed-forward deterministic models often struggle in this ill-posed setting, especially when the inputs are sparse, incomplete, or ambiguous, making it difficult to recover accurate dynamic geometry without strong priors. To address these challenges, we propose *4D Latent Set Diffusion*, a generative method that explicitly learns the probabilistic distribution of deformable surface sequences through both shape and motion diffusion priors. This enables sampling diverse, high-quality object surfaces and motion tracking with multiple plausible outcomes. For compact yet expressive 4D representation, we introduce a latent set-based neural representation equipped with transformer architectures. This design preserves rich geometric details while capturing complex deformations in a low-dimensional latent space.

Consistent with other latent diffusion models, our training consists of two distinct stages: autoencoders and diffusion models. We describe the shape and deformation autoencoders in Section III-A and Fig. 2 *top*. We elaborate the 4D latent set diffusion models in Section III-B and Fig. 2 *bottom*. We also discuss different condition modalities in Section III-C.

A. 4D Neural Representation With Latent Sets

Previous works often utilize single global codes [15], [16], [17] to represent 4D sequences, potentially losing significant surface geometry and temporal evolution details. Instead of a global latent, we assign local latent codes to individual local regions, which significantly improves the network's capability to accurately model non-linear motions and generalize to unseen identities and motions. Given that different non-rigid objects share similar local geometry and deformation patterns, the latent sets can also improve the generalization ability to handle unseen motions and identities. Also inspired by [22], we apply the "VecSet" representation to both the shape and deformation field learning. For a mesh sequence of T frames, the 4D latent set is defined as:

$$\underbrace{\mathbb{R}^{M\times C_s}}_{\text{shape}} \times \underbrace{\mathbb{R}^{(T-1)\times M\times C_d}}_{\text{deformation}}, \tag{2}$$

where M is the size of the latent set, and C_s and C_d are the channels of the shape and deformation latent space, respectively. The *shape latent set* is responsible for reconstructing the first frame, serving as the reference frame. The flow fields are decoded by the *deformation latent sets*, which are used to predict the dense correspondence between the initial frame and the subsequent frames.

1) Shape Latent Set: Following the point query variant of 3DShape2VecSet [22], we compress the 3D surface of the initial frame into a set of latent codes. The input of the autoencoder is a surface point cloud $\mathbf{X}$ of size $N = 2048$. We further downsample the input to a smaller point cloud of size $M = 512$. A cross attention layer is then applied to obtain the compressed

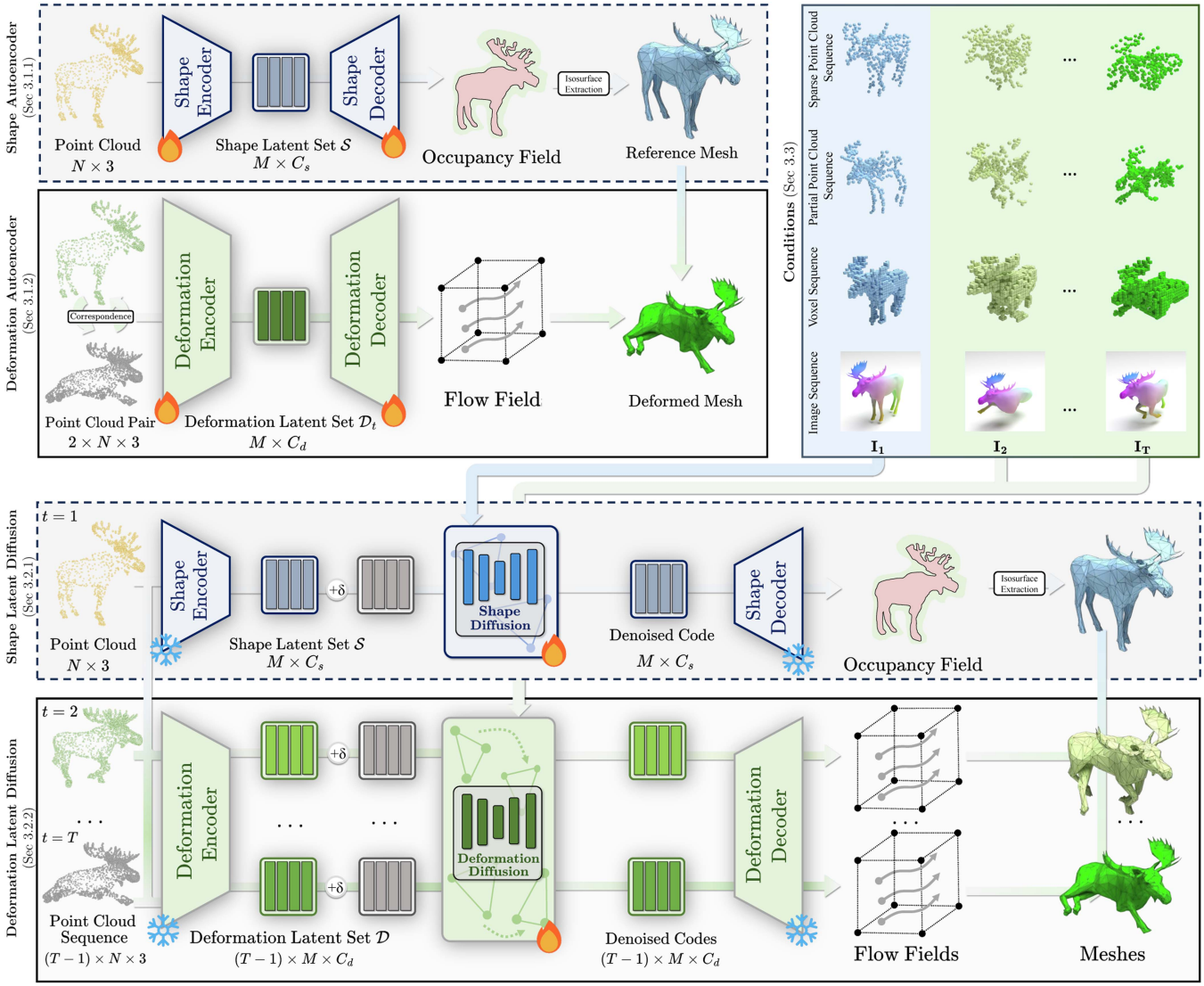


Fig. 2. Overview Pipeline of Motion2VecSets. Our method consists of two stages: 4D latent set representation learning and 4D latent set diffusion. In the first stage, the *Shape Autoencoder* encodes the first reference frame into a shape latent set $\mathcal{S} \in \mathbb{R}^{M \times C_s}$. The *Deformation Autoencoder* encodes each pair of point clouds formed between the first and a subsequent frame into a deformation latent set $\mathcal{D}^t \in \mathbb{R}^{M \times C_d}$. Stacking all $\mathcal{D}^t$ across time yields a motion latent tensor $\mathcal{D} \in \mathbb{R}^{(T-1) \times M \times C_d}$. In the second stage, given a sequence of multi-modal observations $\{\mathbf{I}_t\}_{t=1}^{T}$ (e.g., images, voxels, or point clouds), we use modality-specific encoders to extract conditioning embeddings $\mathcal{C}$. Conditioning on $\mathcal{C}_1$, the *Shape Latent Diffusion* denoises a noisy shape latent to reconstruct the occupancy field of the reference frame via the frozen shape decoder. A surface mesh can be obtained via iso-surface extraction. Conditioning on $\mathcal{C}_2, \ldots, \mathcal{C}_T$, the *Synchronized Deformation Latent Diffusion* jointly denoises the deformation latents for all subsequent frames. The denoised latents are then decoded by the frozen deformation decoder into flow fields, which deform the reference mesh over time.

shape latent $\mathcal{S} \in \mathbb{R}^{M \times C_s}, C_s = 32$. The encoder processes $\mathbf{X}$ to obtain $\mathcal{S}$ as follows:

$$\begin{aligned} \text{index} &= \text{FPS}(\mathbf{X}) && \text{find index} \\ \mathbf{Z} &= \text{PE}(\mathbf{X}) && \text{point embed.} \\ \mathbf{Z}^* &= \text{IndexSelect}(\mathbf{Z}, \text{index}) && \text{subsample point embed.} \\ \mathcal{S} &= \text{KL}(\text{CrossAttn}(\mathbf{Z}, \mathbf{Z}^*)) && \text{compress} \end{aligned} \tag{3}$$

In the decoder, a cross-attention layer is used to fuse the latent codes for occupancy prediction of 3D points through an MLP. To learn the shape latent set, we train the shape auto-encoder by minimizing the binary cross-entropy (BCE) loss between the predicted occupancies of query points randomly sampled in 3D space and actual occupancies:

A2) Deformation Latent Set: To compress temporal shape deformations into compact latent sets, we design a dedicated deformation autoencoder that encodes pairwise deformations between the initial and subsequent frames into downsampled latent sets. To do so, we design a deformation autoencoder which is illustrated in Fig. 3. Unlike Shape Autoencoder that processes single frame, our deformation autoencoder has two input frames: the source $\mathbf{X}_{\text{src}}$ and target point clouds $\mathbf{X}_{\text{tgt}}$ of size $N = 2048$. This pairwise input structure requires a novel

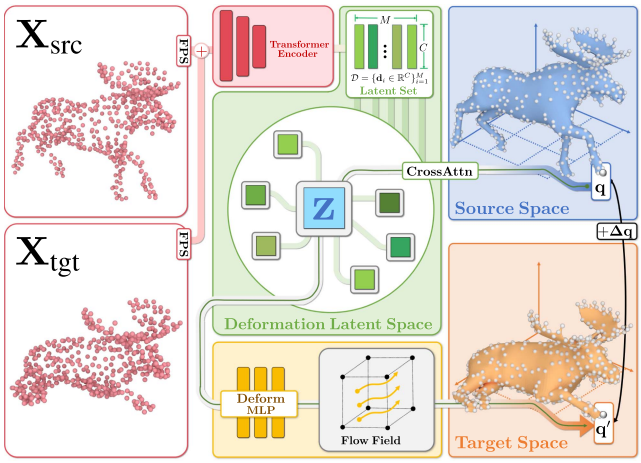


Fig. 3. Deformation Autoencoder. Given a pair of point clouds $\mathbf{X}_{\text{src}}$ and $\mathbf{X}_{\text{tgt}}$ from two frames of a dynamic mesh sequence, we initially downsample them using farthest point sampling (FPS). Subsequently, the concatenated points are fed into transformer encoder to generate the Deformation Latent Set $\mathcal{D}$. For a query point $\mathbf{q}$ in the source space, a cross-attention layer is utilized to retrieve the most relevant fused feature $\mathbf{z}$. This selected feature is subsequently fed into the deformation MLP decoder to predict an offset $\Delta\mathbf{q}$, which translates it to $\mathbf{q}'$ in the target space. To reduce the feature diversity of $\mathcal{D}$, KL-regularization is employed.

encoder architecture, described as follows:

$$\begin{aligned}
\text{index} &= \text{FPS}(\mathbf{X}_{\text{src}}) && \text{find index}\\
\mathbf{Z}_{\text{src}} &= \text{PE}(\mathbf{X}_{\text{src}}) && \text{source embed}\\
\mathbf{Z}_{\text{tgt}} &= \text{PE}(\mathbf{X}_{\text{tgt}}) && \text{target embed}\\
\mathbf{Z}^*_{\text{src}} &= \text{IndexSelect}(\mathbf{Z}_{\text{src}}, \text{index}) && \text{subsample source embed}\\
\mathbf{Z}^*_{\text{tgt}} &= \text{IndexSelect}(\mathbf{Z}_{\text{tgt}}, \text{index}) && \text{subsample target embed}\\
\mathbf{Z} &= \text{Concat}([\mathbf{Z}_{\text{src}}, \mathbf{Z}_{\text{tgt}}], -1) && \text{concat along channel}\\
\mathbf{Z}^* &= \text{Concat}([\mathbf{Z}^*_{\text{src}}, \mathbf{Z}^*_{\text{tgt}}], -1) && \text{concat along channel}\\
\mathcal{D} &= \text{KL}(\text{CrossAttn}(\mathbf{Z}, \mathbf{Z}^*)) && \text{compress}
\end{aligned} \tag{4}$$

We begin by applying Farthest Point Sampling (FPS) to the source point cloud to obtain a set of downsampled indices. These indices are then propagated to the target point cloud using IndexSelect$(\cdot,\cdot)$, thereby preserving inter-frame correspondences. This operation ensures that features are extracted from consistent local surface regions across frames. While one could employ cross-attention layers to align and fuse features between the source and target through explicit correspondence search, we adopt this simpler yet effective strategy to capture deformation patterns accurately. This design choice not only learns accurate deformation features but also avoids the computational overhead associated with attention-based matching. Similar to the shape autoencoder, we employ a cross-attention mechanism CrossAttn$(\cdot,\cdot)$ alongside a KL divergence regularizer KL$(\cdot)$ to obtain compressed deformation latents $M \times C_d, C_d = 32$, effectively capturing localized surface deformation patterns from the source to the target frame. In the decoder, we again leverage cross-attention to reconstruct the flow field, which deforms the source shape that approximates the target shape. The reconstruction loss is defined as the ℓ_2 distance between the predicted and ground-truth target point clouds.

To ensure that deformation latents consistently represent the same local surfaces or object parts across time, we reuse the FPS downsampling indices from the initial frame for all subsequent frames. This design enables direct stacking of deformation latents along the temporal dimension. Benefiting from the correspondence-preserving property of our deformation autoencoder, we can apply attention operations across frames to exchange deformation information, thereby enhancing temporal coherence. This temporal alignment is also crucial for the efficiency of our denoising network, as discussed later in Section III-B. With a mild notational simplification, we denote the resulting motion latent for a sequence as $\mathcal{D} \in \mathbb{R}^{(T-1)\times M\times C_d}$, obtained by stacking the per-frame deformation latents.

. *4D Latent Set Diffusion*

1) Shape Diffusion: Following the diffusion paradigm in EDM by Karras et al. [104], we aim to minimize the expected ℓ_2-denoising error. This is achieved by adding the noise $\boldsymbol{\epsilon}$ sampled from the Gaussian distribution to the shape latent set $\mathcal{S}$, and then feeding the noise-added code $\hat{\mathcal{S}} = \mathcal{S} + \boldsymbol{\epsilon}$ to the denoiser (to avoid confusing, we also use $\mathcal{S}$ to represent its matrix form $\mathbb{R}^{N\times C_s}$). The whole process is denoted as:

$$\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\sigma^2\mathbf{I})} \left\| \text{ShapeDenoiser}\left(\hat{\mathcal{S}}, \sigma, \mathcal{C}\right) - \mathcal{S} \right\|_2^2 \tag{5}$$

Here, σ represents the noise level. $\mathcal{C}$ is the conditioning latents extracted from the first input frame $\mathbf{I}_1$ which can be a voxel grid, a point cloud, or an image (see Section III-C).

2) Synchronized Deformation Diffusion: To adapt these 3D models [11], [12], [13], [14] directly to 4D, the most straightforward approach is frame-by-frame processing, which may lead to discontinuities in temporal and spatial correspondence. Another approach is to aggregate all spatial-temporal latents, which would significantly increase the time complexity to $O(T^2{\sim}M^2)$ for a sequence of T frames and M deformation latents. However, our 4D latent set representation allows us to bypass the need for full attention across spatial and temporal domains. As discussed in Section III-A, the deformation latents at the same indices across different frames correspond to the deformation behaviors of the same local surface region. Leveraging this property, we implement an alternating way for latent feature aggregation, systematically switching between the spatial and temporal domains. This method not only preserves the spatio-temporal consistency, but also reduces the computational complexity to $O(TM^2)$ in the spatial domain and $O(MT^2)$ in the temporal domain. The details of synchronized deformation diffusion are described as follows. Given a sequence of input observations $\mathcal{I} = \{\mathbf{I}_t\}_{t=1}^T$, we pair subsequent frames with the first frame, *i.e.*, $\{\mathbf{I}_1, \mathbf{I}_t\}_{t=2}^T$. These pairs are encoded into a series of conditional latents $\mathcal{C}_t = \{\mathbf{c}_i \in \mathbb{R}^C\}_{i=1}^L, C = 32$ via conditioning networks which will be described in Section III-C. Then these conditional latents, together with the diffused shape latent set $\mathcal{S}$ in Section III-B1, are injected into the denoising network as the condition providing

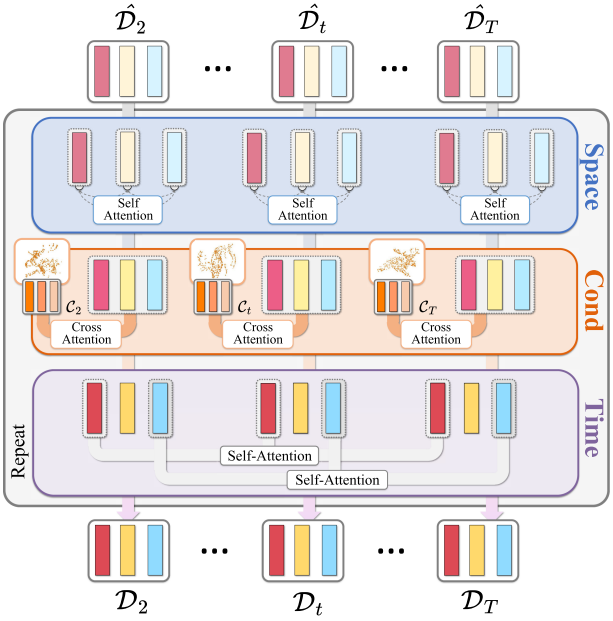


Fig. 4. Synchronized Deformation Set Diffusion. Given noised deformation vector sets $\{\hat{\mathcal{D}}_t\}_{t=2}^T$ (top) from a sequence, each set denoted as $\hat{\mathcal{D}}_t = \{\hat{\mathbf{d}}_t^1, \ldots, \hat{\mathbf{d}}_t^M\}$ of timestep $t \in [2, T]$, we use repeated Interleaved Spatio-Temporal Attention Blocks (ISTA) as our denoising network. In each ISTA block, we first pass them to the space self-attention layer (Space Attention) to aggregate latent features $\hat{\mathcal{D}}^t$ across different spatial locations within each frame to explore spatial contexts. Next, we inject conditional information extracted from imperfect inputs via cross-attention (Condition Attention) between conditional codes $\mathcal{C}_t$ and noised deformation codes $\hat{\mathcal{D}}_t$ at each frame. The inputs could be sparse or partial point clouds, images, or voxel grids. Here we use point clouds as an example. Lastly, to enhance temporal coherence, a time self-attention layer (Time Attention) is used to aggregate latent codes from the same position but from different frames, i.e. $\{\hat{\mathbf{d}}_t^i\}_{t=2}^T$. Repeat this ISTA block and we finally get denoised deformation latent sets $\{\mathcal{D}_t\}_{t=2}^T$ (bottom). Within each layer, different colored latents represent the dynamics of distinct local regions, while the same colored latents represent the dynamics of a local region at different time steps.

guidance for the network to handle ambiguous inputs, like partial point clouds.

Interleaved Spatio-Temporal Attention: Fig. 4 depicts the denoiser network of our proposed synchronized deformation latent set diffusion. The basic unit is the designed Interleaved Spatio-temporal Attention Block (ISTA). We first linearly project the shape latents (with channel dimension C_s) and deformation latents (with channel dimension C_d) into a shared embedding space of dimension C. This results in the following inputs to the denoising network: noisy motion latents of shape $[B, T-1, M, C]$, shape latents of shape $[B, M, C]$, and conditioning latents of shape $[B, T-1, L, C]$, where B is the batch size. Each Interleaved Spatio-Temporal Attention (ISTA) block consists of three attention layers:

- *Space Self-Attention Layer:* Performs self-attention along the spatial dimension (M). To enable this, the input tensor is reshaped to $[B \times (T-1), M, C]$, allowing the network to model spatial dependencies within each frame independently.
- *Conditional Cross-Attention Layer:* This layer injects conditioning signals into the denoising network via cross-attention. Prior to the attention computation, both the input deformation latents and conditioning latents are reshaped to $[B \times (T-1), L, C]$. In addition to the primary conditioning, we further incorporate shape latents $\mathcal{S}$ by applying cross-attention between $\mathcal{S}$ and the deformation features of each temporal frame. It enables the network to modulate the motion dynamics based on the underlying static geometry.
- *Time Self-Attention Layer:* This layer performs self-attention along the temporal axis ($T-1$), allowing the network to aggregate motion features over time. To enable this, the input is reshaped to $[B \times M, T-1, C]$ before applying the attention operation. By attending across frames for each spatial location, this layer effectively captures temporal dependencies and consolidates deformation features of corresponding local regions throughout the sequence.

In the denoising phase, we regard the entire sequence of deformation codes as a motion latent set and jointly denoise them together to learn temporal motion priors. We add a Gaussian noise $\boldsymbol{\epsilon}$ to the motion latent set $\mathcal{D}$ to obtain its noise-corrupted version $\hat{\mathcal{D}}$. The denoising objective is thus formulated as:

$$\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\sigma^2\mathbf{I})} \left\| \text{MotionDenoiser}\left(\hat{\mathcal{D}}, \sigma, \mathcal{S}, \mathcal{C}\right) - \mathcal{D} \right\|_2^2 \quad (6)$$

Here, $\mathcal{C}$ is the conditioning latents $\{\mathcal{C}_2, \mathcal{C}_3, \ldots, \mathcal{C}_T\}$.

C. Conditioning Network

In this section, we describe how we extract conditioning embeddings $\mathcal{C}$ from ambiguous inputs $\mathcal{I}$, such as point clouds, voxel grids, images, and sparse handle movements. Given *raw* conditioning signal $\mathcal{I}$ of T frames

$$\begin{aligned} \{\tilde{\mathcal{C}}_1, \ldots, \tilde{\mathcal{C}}_T\} &= \text{Enc}(\mathcal{I}_1, \ldots, \mathcal{I}_T), && \text{per frame embeds} \\ C_1 &= \tilde{\mathcal{C}}_1, \\ C_2 &= \text{Concat}([\tilde{\mathcal{C}}_1, \tilde{\mathcal{C}}_2], -1), && \text{merge with ref} \\ &\vdots \\ C_T &= \text{Concat}([\tilde{\mathcal{C}}_1, \tilde{\mathcal{C}}_T], -1), && \text{merge with ref.} \end{aligned} \quad (7)$$

For reference shape generation in Section III-B1, we condition the network on $\mathcal{C}_1$. For motion generation in Section III-B2, we condition on $\mathcal{C}_t$, which represents the relative deformation features of frame t with respect to the first reference frame.

Point clouds: The input point clouds ($L = 300$ or 512 points) may be sparse, partial, or noisy. We encode them into feature embeddings using a transformer encoder, which can obtain L latents.

Voxel grids: A voxel grid represents 3D shapes using binary occupancy values. We process voxel grids at a resolution of 32^3 using a series of 3D convolutional layers to extract volumetric features of resolution 8^3, which are then flattened across spatial dimensions into a sequence of $L = 512$ latents.

RGB images: We use RGB images of resolution $224 \times 224 \times 3$ as inputs. Each image is processed using a pretrained DINO-v2 ViT-B/14 backbone [105], which encodes it into a feature map of size $16 \times 16 \times 768$. This feature map is then flattened into a

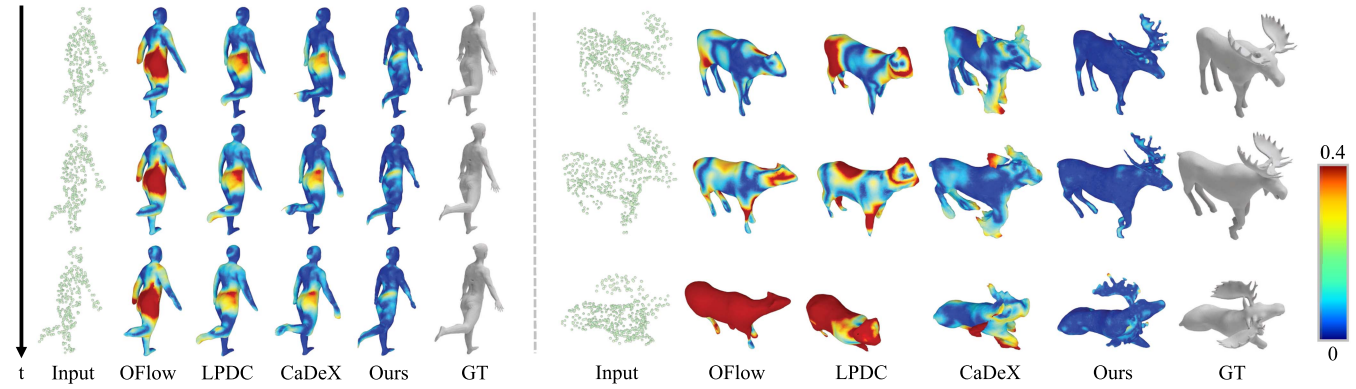


Fig. 5. Qualitative comparisons of 4D shape reconstruction from **sparse and noisy** point clouds on the D-FAUST [25] (left) and DT4D-A [26] (right) datasets. We visualize the Chamfer Distance between reconstruction and ground truth as error maps. Our method reconstructs more accurate surface geometries and motion dynamics.

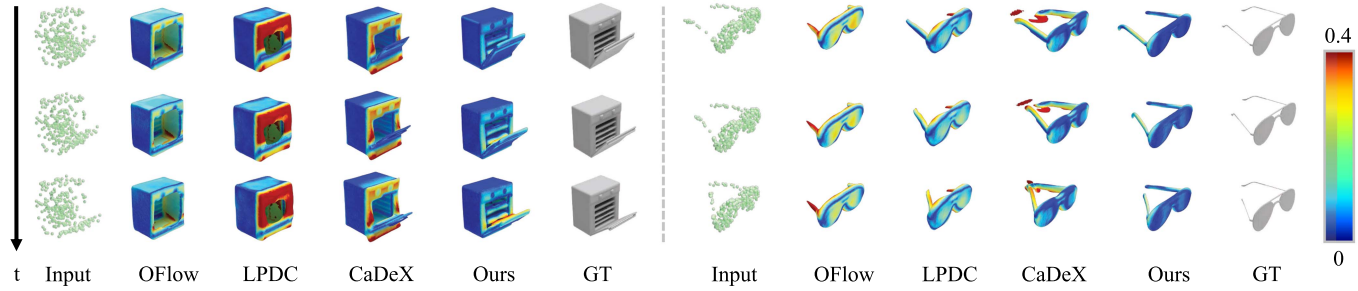


Fig. 6. Qualitative comparisons of 4D shape reconstruction from **sparse and noisy** point clouds on the S2M [106] dataset. Our method reconstructs more accurate surface geometries and motion dynamics.

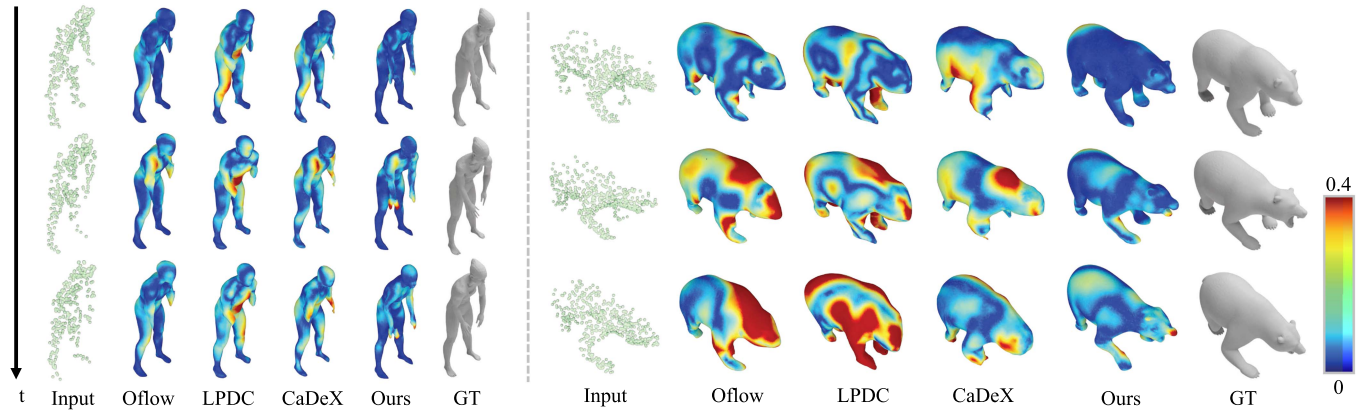


Fig. 7. Qualitative comparisons of 4D shape completion from **monocular noisy depth scans** on the D-FAUST [25] (left) and DT4D-A [26] (right) datasets. Our method reconstructs more accurate surface geometries and motion dynamics.

sequence of $L = 257$ latents, comprising 256 patch embeddings and one global embedding.

IV. Experiments

Datasets: We conducted experiments on three 4D datasets. The first, the Dynamic FAUST (D-FAUST) [25], focuses on human body dynamics, including 10 subjects and 129 sequences. It is split into training (70%), validation (10%), and test (20%) subsets, following [15]. The second, the DeformingThings4D-Animals (DT4D-A) [26], includes 38 identities with a total of 1227 animations, divided into training (75%), validation (7.5%), and test (17.5%) subsets following [17]. The training and validation sets use motion sequences of seen individuals. The test set is divided into two parts: unseen motions and unseen individuals. The last dataset, the Shape2Motion (S2M) [106], consists of articulated objects across 8 diverse categories, including laptops, doors, staplers, eyeglasses, washing machines, refrigerators, and

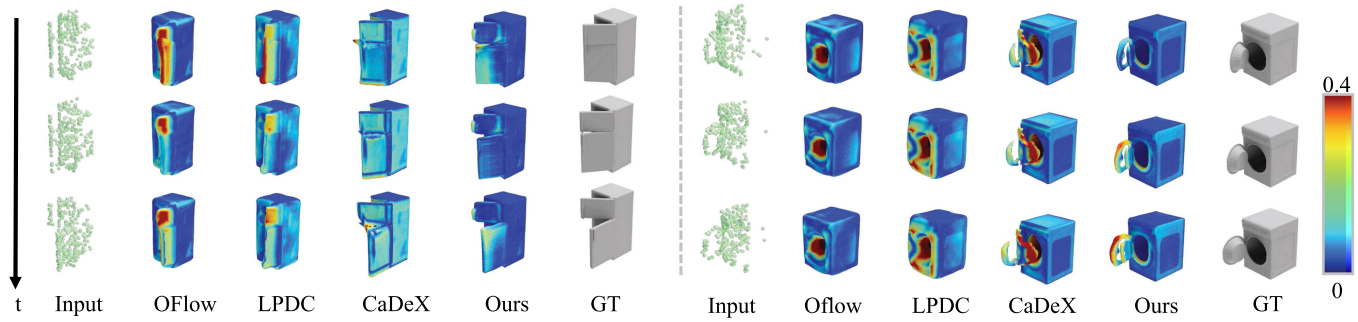


Fig. 8. Qualitative comparisons of 4D shape completion from **monocular noisy depth scans** on the S2M [106] dataset. We visualize the Chamfer Distance between the reconstruction and the ground truth as error maps. Our method exhibits lower reconstruction errors and achieves more plausible motion tracking.

ovens. Each object category includes one or more articulation controls. For this dataset, we follow the experimental setup proposed by CaDeX [17] and A-SDF [27], where the articulation is rotated to generate 30 frames per object. Among all generated sequences within a given category, 80% are selected as training objects, while the remaining sequences are allocated to the test split.

Baselines: We compare against state-of-the-art methods in 4D reconstruction, including OFlow [15], LPDC [16], CaDex [17]. **OFlow** assigns each 4D point both an occupancy value and a motion velocity vector, utilizing a Neural-ODE framework [107] for learning deformations. **LPDC** employs an MLP to parallelly learn correspondences among occupancy fields across different time steps via explicitly learning continuous displacement vector fields from spatio-temporal shape representation. **CaDeX** introduces a canonical map factorization and utilizes invertible deformation networks to maintain homeomorphisms. For fair comparisons, we follow their original training paradigms.

Evaluation Metrics: The Intersection over Union (IoU) evaluates the volume overlap between predicted and ground truth meshes. The Chamfer Distance (CD) calculates the average nearest-neighbor distance between two point sets. ℓ_2-distance measures the Euclidean distance between corresponding points on the predicted and ground truth meshes.

Implementations: Our training pipeline consists of two stages. In the first stage, we train the shape and deformation auto-encoders. For the shape auto-encoder, we use a learning rate of 10^{-4} and a KL-divergence loss weight of 10^{-3}. For the deformation auto-encoder, the learning rate is also 10^{-4}, with a KL-divergence loss weight of 10^{-6}. Both models are trained for 100 epochs with a batch size of 24. In the second stage, we train the diffusion models. The learning rate is set to 10^{-4} for both the shape and deformation diffusion networks. We train the shape diffusion model for 50 epochs with a batch size of 8, and the deformation diffusion model with a batch size of 4. Both the shape and deformation diffusion models follow the noise scheduling strategy of EDM [104]. During training, the noise level σ is sampled from a log-normal distribution $\mathcal{N}(\mu = -1.2, \sigma = 1.2)$. During inference, we use 18 denoising steps with noise levels linearly scheduled from $\sigma_{\text{max}} = 80$ to $\sigma_{\text{min}} = 0.002$. Note that the shape and deformation auto-encoders can be trained in parallel, as can the shape and deformation diffusion models. The shape and deformation autoencoders are individually trained for approximately 10 hours, while the shape and deformation diffusion models require around 20 and 30 hours, respectively. All models are trained on two NVIDIA H200 GPUs.

TABLE II
QUANTITATIVE COMPARISONS OF 4D SHAPE RECONSTRUCTION FROM **SPARSE AND NOISY** POINT CLOUD SEQUENCES ON THE DT4D-A [26], D-FAUST [25], AND S2M [106] DATASETS

Dataset	Method	Unseen Motion			Unseen Individual		
		IoU ↑	CD ↓	Corr ↓	IoU ↑	CD ↓	Corr ↓
DT4D-A [26]	OFlow [15]	70.6%	0.104	0.204	57.3%	0.175	0.285
	LPDC [16]	72.4%	0.085	0.162	59.4%	0.149	0.262
	CaDex [17]	80.3%	0.061	0.133	64.7%	0.127	0.239
	Ours	**88.9%**	**0.050**	**0.061**	**83.7%**	**0.058**	**0.074**
D-FAUST [25]	OFlow [15]	81.5%	0.065	0.094	72.3%	0.084	0.117
	LPDC [16]	84.9%	0.055	0.080	76.2%	0.071	0.098
	CaDex [17]	89.1%	0.039	0.070	80.7%	0.055	0.087
	Ours	**90.7%**	**0.033**	**0.047**	**83.7%**	**0.045**	**0.064**
S2M [106]	OFlow [15]	/	/	/	53.2%	0.223	0.204
	LPDC [16]	/	/	/	56.7%	0.173	0.163
	CaDeX [17]	/	/	/	58.9%	0.118	0.160
	Ours	/	/	/	**75.9%**	**0.070**	**0.095**

A. 4D Shape Reconstruction From Point Clouds

We initially assessed our model's ability for 4D reconstruction from sparse and noisy point cloud sequences Following the setup in OFlow [15], our network processed sequences of $T = 17$ continuous frames.

1) Sparse & Noisy Point Cloud Sequences: Each frame represents a sparse point cloud, with $L = 300$ for D-FAUST [25] and S2M [106] or $L = 512$ for DT4D-A [26]. We also simulate noisy observations by adding Gaussian noise ($\sigma = 0.05$).

Quantitatively, our model consistently outperforms previous methods on the D-FAUST [25], DT4D-A [26], and S2M [106] datasets, as shown in Table II. The performance gain is especially notable on the unseen individual split of DT4D-A, which features diverse topologies across various animal species. Our method improves IoU by 19% and 17% over the previous best models on DT4D-A and S2M, respectively. Additionally, it reduces both Chamfer Distance and ℓ_2-correspondence error to less than half of those reported by existing approaches on D-FAUST and DT4D-A.

TABLE III
QUANTITATIVE COMPARISONS OF 4D SHAPE COMPLETION FROM **MONOCULAR NOISY DEPTH SCANS** ON THE DT4D-A [26], D-FAUST [25], AND S2M [106] DATASETS

Dataset	Method	Unseen Motion			Unseen Individual		
		IoU ↑	CD ↓	Corr ↓	IoU ↑	CD ↓	Corr ↓
DT4D-A [26]	OFlow [15]	64.2%	0.305	0.423	55.1%	0.408	0.538
	LPDC [16]	62.2%	0.339	0.427	51.6%	0.467	0.488
	CaDex [17]	70.8%	0.254	0.499	59.2%	0.379	0.498
	Ours	**73.3%**	**0.177**	**0.404**	**66.3%**	**0.193**	**0.438**
D-FAUST [25]	OFlow [15]	76.9%	0.084	0.165	66.4%	0.109	0.194
	LPDC [16]	68.3%	0.138	0.167	59.6%	0.156	0.204
	CaDex [17]	80.7%	0.074	0.123	70.4%	0.096	0.157
	Ours	**83.8%**	**0.054**	**0.111**	**74.4%**	**0.075**	**0.140**
S2M [106]	OFlow [15]	/	/	/	53.6%	0.232	0.228
	LPDC [16]	/	/	/	54.5%	0.217	0.196
	CaDeX [17]	/	/	/	56.3%	0.145	0.186
	Ours	/	/	/	**63.6%**	**0.128**	**0.175**

Qualitatively, as shown in Figs. 5 and 6, our model excels at reconstructing complete shapes with minimal Chamfer Distance errors. This advantage is especially clear in challenging regions such as fast-moving body parts (e.g., human feet), highly articulated structures (e.g., animal heads), and fine details (e.g., eyeglass temples).

Our model's strength lies in the 4D latent set diffusion, which enables accurate sampling of local geometry and deformation patterns. While methods like LPDC [16] and OFlow [15] perform well on human datasets with consistent topology, they struggle with the diverse shapes and scales in animal data. Global latent-based approaches also fail to reliably track fast-moving articulated parts, such as hinges. In contrast, our method effectively models complex 4D dynamics across a wide range of non-rigid objects.

2) Monocular Depth Sequences: To simulate sparse and partial real-world scans, we generated monocular depth sequences by rendering from a fixed camera angle. The size of the input point cloud and the frame length are the same as Section IV-A1.

Quantitative results in Table III demonstrate that on the S2M [106] dataset, our method achieves an IoU of 63.6%, outperforming the strongest baseline (CaDeX) by 7.3%, and reducing the Chamfer Distance from 0.145 to 0.128. Similar improvements are observed on the DT4D-A [26] dataset, where our method improves IoU by 7.1% over baselines for unseen individuals. Even on relatively simpler datasets such as D-FAUST [25], our method consistently improves IoU by 4.0% over CaDeX and further reduces the Chamfer Distance.

Qualitative results in Figs. 7 and 8 further verify our quantitative findings. Baseline methods often produce fragmented surfaces and exhibit inconsistent deformations, particularly in human feet or animal heads. In contrast, our method generates more complete and temporally consistent reconstructions. This advantage is more obvious on the S2M [106] dataset, where our approach preserves fine-scale structural details across time for articulated objects.

The comparisons show that our method reconstructs more complete surfaces with more accurate motion tracking, highlighting the effectiveness of our 4D latent set diffusion in handling ambiguous inputs such as partial scans.

TABLE IV
QUANTITATIVE COMPARISONS OF 4D SHAPE SUPER-RESOLUTION FROM **VOXEL GRID SEQUENCES** WITH RESOLUTION 32^3 ON THE DT4D-A [26] DATASET

Input	Method	Unseen Motion			Unseen Individual		
		IoU ↑	CD ↓	Corr ↓	IoU ↑	CD ↓	Corr ↓
4D Voxel of DT4D-A	OFlow [15]	68.4%	0.237	0.385	58.6%	0.311	0.412
	LPDC [16]	69.3%	0.231	0.311	59.2%	0.321	0.343
	CaDeX [17]	77.1%	0.152	0.269	65.8%	0.228	0.331
	Ours	**84.3%**	**0.066**	**0.126**	**75.6%**	**0.088**	**0.187**

TABLE V
QUANTITATIVE COMPARISONS OF 4D SHAPE RECONSTRUCTION FROM **RGB IMAGE SEQUENCES** ON THE D-FAUST [25] DATASET

Input	Method	Unseen Motion			Unseen Individual		
		IoU ↑	CD ↓	Corr ↓	IoU ↑	CD ↓	Corr ↓
Image sequence of D-FAUST	OFlow [15]	51.5%	0.284	0.319	31.0%	0.431	0.442
	LPDC [16]	52.3%	0.255	0.282	35.2%	0.371	0.397
	CaDeX [17]	53.8%	0.229	0.284	38.7%	0.288	0.352
	Ours	**69.9%**	**0.119**	**0.206**	**51.6%**	**0.189**	**0.299**

B. 4D Shape Super-Resolution From Voxel Grids

We also conduct comparisons on the task of 4D shape super-resolution from a sequence of coarse voxel grids on the DT4D-A [26] dataset. This task poses significant challenges due to spatial quantization and limited geometric details in the low-resolution volumetric grids. For all baselines, we re-implement all baselines with global latent codes derived from volumetric features as conditions. Quantitative results in Table IV show that our method outperforms all baselines on the DT4D-A [26] dataset. It achieves an IoU of 84.3%, surpassing CaDeX [17] by over 7%, and reduces the Chamfer Distance by nearly 50%. Qualitative results are shown in Fig. 9, where baseline methods often produce fragmented surfaces and suffer from temporal inconsistencies in complex animal motions. In contrast, our model generates smoother temporal evolution and more complete structures. Remarkably, it can even hallucinate plausible details, such as continuous antlers or smooth limb connections, which are absent in the coarse voxel inputs, highlighting the strength of our diffusion-based 4D surface priors.

C. 4D Shape Reconstruction From RGB Images

We also evaluate our method on the task of 4D shape reconstruction from RGB image sequences rendered from the D-FAUST [25] dataset. This setting poses greater challenges than voxel or point cloud inputs, as it lacks explicit 3D geometry and relies solely on 2D visual cues. Quantitative results in Table V show that on unseen motions, our method improves IoU by over 16% compared to the strongest baseline, CaDex, while also significantly reducing Chamfer Distance and ℓ_2-correspondence errors. As shown in Fig. 10, existing methods often generate incomplete surfaces and exhibit temporal instability, especially around articulated limbs, failing to track motion consistently. However, our model produces coherent shape sequences with improved structural integrity and smooth temporal evolution. Overall, these comparisons demonstrate that our approach enables accurate and temporally consistent 4D reconstruction directly from RGB image sequences.

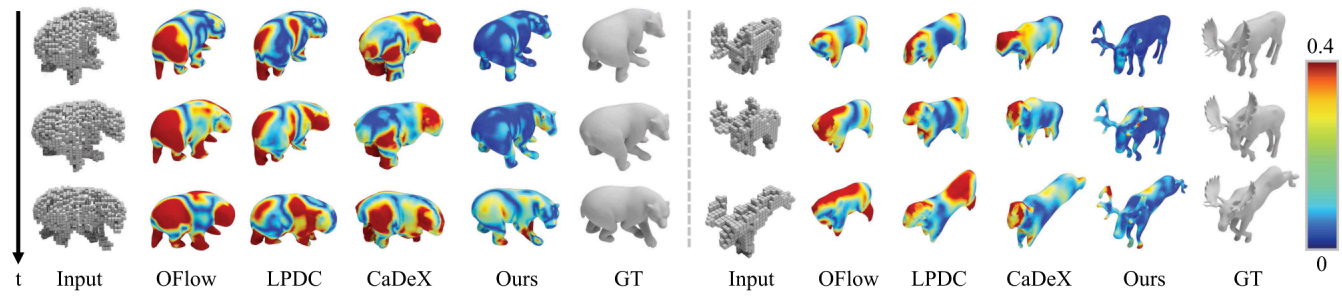


Fig. 9. Qualitative comparisons of 4D shape super-resolution from **voxel grid sequences** with resolution 32^3 on the DT4D-A [26] dataset. Our method exhibits lower reconstruction errors and achieves more plausible tracking.

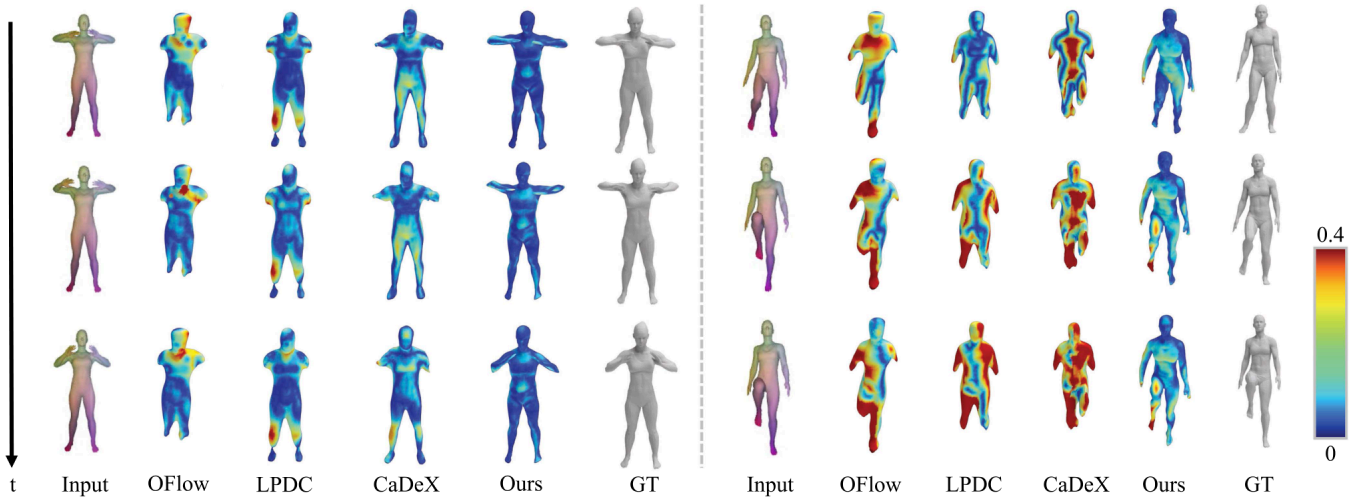


Fig. 10. Qualitative comparisons of 4D shape completion from **RGB image sequences** on the D-FAUST [25] dataset. Our method exhibits lower reconstruction errors and achieves more plausible tracking.

TABLE VI
QUANTITATIVE ABLATION STUDIES OF 4D SHAPE COMPLETION FROM MONOCULAR NOISY DEPTH SCANS ON THE D-FAUST [25] DATASET. M DENOTES THE NUMBER OF LATENT CODES AND C_d REPRESENTS THE NUMBER OF LATENT CODE CHANNELS.

Method	Unseen Motion			Unseen Individual		
	IoU↑	CD↓	Corr↓	IoU↑	CD↓	Corr↓
W/o. Diffusion	71.1%	0.097	0.173	64.2%	0.107	0.194
$M = 1$	68.5%	0.120	0.301	57.7%	0.149	0.327
$C_d = 8$	78.9%	0.078	0.180	68.0%	0.105	0.225
$C_d = 16$	78.0%	0.080	0.189	66.8%	0.109	0.254
W/o. TimeAttn.	81.2%	0.061	0.127	70.8%	0.086	0.158
Full ($M = 512$, $C_d = 32$)	**83.8%**	**0.054**	**0.111**	**74.4%**	**0.075**	**0.140**

D. Ablation Study

We conducted ablation studies to validate the effectiveness of each component (see Table VI, Fig. 11) under the setting of 4D shape completion from **monocular noisy depth scans** on the D-FAUST [25] dataset.

What is the effect of diffusion model? 4D surface reconstruction from ambiguous observations of noisy, sparse, or partial point clouds is an ill-posed problem. Deterministic models often yield suboptimal results. We compare against a feedforward baseline that uses the same autoencoder architecture but without diffusion. Specifically, it takes as input a sequence of partial point clouds (300 points per frame) and directly predicts the full surfaces in a single forward pass. As shown in Fig. 11 and Table VI, the diffusion model adopts a probabilistic way to deal with highly ambiguous inputs and generates plausible predictions. Moreover, diffusion models can handle "one-to-many" problems and generate diverse and creative outputs as shown in Fig. 16.

What is the effect of 4D latent set representation? Instead of using a single global latent code, our approach employs 4D latent sets. Both variants share the same encoder-decoder backbone to ensure a fair comparison. As indicated in Table VI, our method significantly outperforms the global latent codes (with $M = 1$) and captures more accurate 4D motions. This advantage becomes more apparent for unseen identities, demonstrating better generalization ability.

What is the effect of time attention layers? For the synchronized deformation latent set diffusion, we integrated the time self-attention layer in our interleaved spatio-temporal attention mechanism. Removing this layer decreased all metrics, highlighting its effectiveness in maintaining temporal coherence.

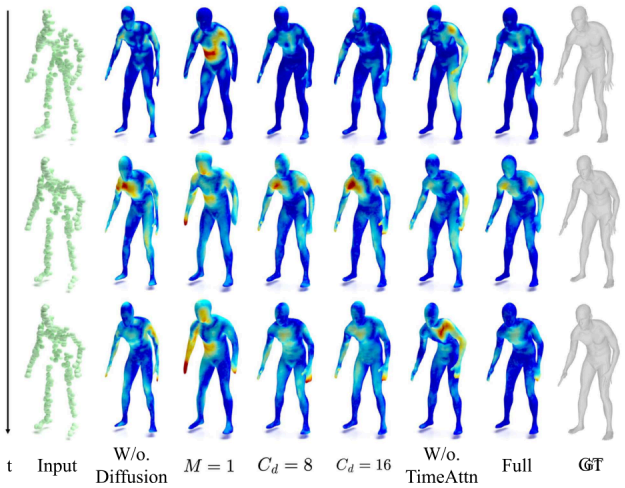


Fig. 11. **Qualitative ablation studies** of 4D shape completion from **monocular noisy depth scans** on the D-FAUST [25] dataset. Without diffusion, reconstructions suffer from incomplete geometry due to the ill-posed nature of sparse inputs. Using a single global latent ($M = 1$) or fewer channels ($C_d = 8/16$) limits the ability to capture local deformations. Removing temporal attention introduces motion discontinuities. The full model with diffusion, 4D latent sets ($M = 512$), and temporal attention achieves the most coherent and accurate 4D reconstructions.

What is the effect of the number of channels of latent set? For the shape latent set, we follow 3DShape2VecSet [22] and set the number of shape channels to $C_s = 8$. To determine the optimal number of deformation latent channels C_d for learning deformation priors on time-varying surfaces, we conduct an ablation study while keeping C_s fixed. As shown in Table VI, setting $C_d = 32$ provides more favorable performance for 4D latent set diffusion.

What is the effect of local latent codes? To highlight the advantage of local latent codes in our 4D latent set representation for modeling complex geometries and motions, we conduct additional comparisons on the DT4D-A dataset. We compare (c) our full model (local shape and deformation latents) against two variants: (a) a global shape latent with global deformation latents, and (b) local shape latents with global deformation latents, on the task of 4D shape reconstruction from noisy and sparse point clouds. As shown in the first row of Fig. 12, local shape latents reconstruct complex shapes of unseen individuals even at the first frame, whereas a single global shape latent fails to produce plausible shapes. In the 2nd and 3 rd rows, local deformation latents track more accurate non-linear motions than global deformation latents. These observations are also supported by the quantitative results in Table VII.

E. Cross-Dataset Generalization

To evaluate cross-dataset generalization, we train all methods on DT4D-Animals and test them on the D-FAUST human dataset for 4D shape reconstruction from noisy and sparse point clouds. As shown in Fig. 13, our method remains stable in both geometry reconstruction and motion tracking, while the baseline methods fail to generalize across datasets. As reported in Table VIII, our approach outperforms all baselines across all listed metrics

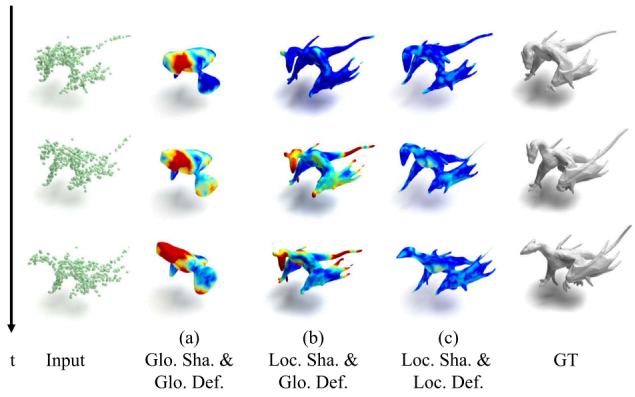


Fig. 12. **Qualitative ablation studies of local latent codes** on the DT4D-A [26] dataset. Given a sequence of noisy and sparse point clouds, we compare the reconstruction results between three method variants: (a) global shape & deformation latents; (b) local shape & global deformation latents; (c) local shape & deformation latents (Our final).

TABLE VII
QUANTITATIVE ABLATION STUDIES OF LOCAL LATENT CODES ON THE DT4D-A [26] DATASET. GIVEN A SEQUENCE OF NOISY AND SPARSE POINT CLOUDS, WE COMPARE THE RECONSTRUCTION RESULTS BETWEEN THREE METHOD VARIANTS: (A) GLOBAL SHAPE & DEFORMATION LATENTS; (B) LOCAL SHAPE & GLOBAL DEFORMATION LATENTS; (C) LOCAL SHAPE & DEFORMATION LATENTS (OUR FINAL).

Method	Unseen Motion			Unseen Individual		
	IoU ↑	L1 ↓	L2 ↓	IoU ↑	L1 ↓	L2 ↓
(a) Glo. Sha. & Glo. Def.	66.8%	0.336	0.378	57.0%	0.395	0.491
(b) Loc. Sha. & Glo. Def.	83.0%	0.116	0.228	73.1%	0.179	0.397
(c) Loc. Sha. & Loc. Def.	**88.9%**	**0.050**	**0.061**	**83.7%**	**0.058**	**0.074**

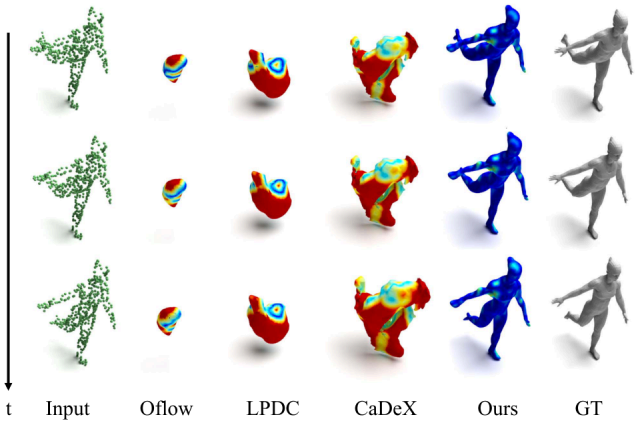


Fig. 13. Qualitative comparisons of **Cross-Dataset Generalization** on the task of 4D shape reconstruction from noisy and sparse point clouds. All methods are trained on the DT4D dataset and directly evaluated on the D-FAUST dataset.

TABLE VIII
QUANTITATIVE COMPARISONS OF **CROSS-DATASET GENERALIZATION** ON THE TASK OF 4D SHAPE RECONSTRUCTION FROM NOISY AND SPARSE POINT CLOUDS. ALL METHODS ARE TRAINED ON THE DT4D DATASET AND DIRECTLY EVALUATED ON THE D-FAUST DATASET.

Dataset	Method	Test Set 1			Test Set 2		
		IoU ↑	L1 ↓	L2 ↓	IoU ↑	L1 ↓	L2 ↓
D-FAUST [25]	OFlow [15]	29.8%	0.427	0.499	24.5%	0.466	0.533
	LPDC [16]	27.1%	0.437	0.527	23.1%	0.430	0.509
	CaDeX [17]	37.6%	0.312	0.405	28.8%	0.337	0.474
	Ours	**82.0%**	**0.059**	**0.078**	**79.9%**	**0.055**	**0.073**

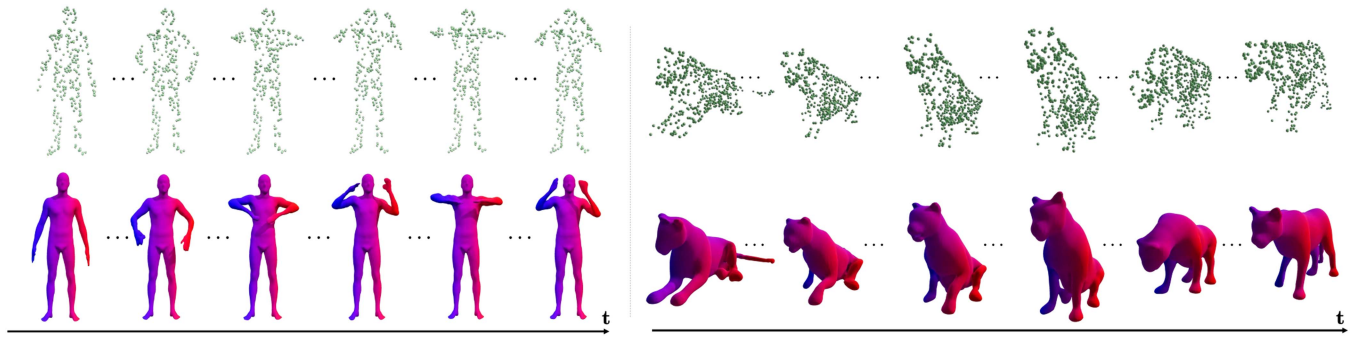


Fig. 14. 4D shape reconstruction on 100-frame sequences from D-FAUST [25] and DT4D-A [26] using sparse and noisy inputs. Although trained only on 17-frame clips, our method can be extended to much longer sequences by performing 17-frame diffusion in an autoregressive manner, maintaining temporal coherence and preserving fine geometric details over extended motions.

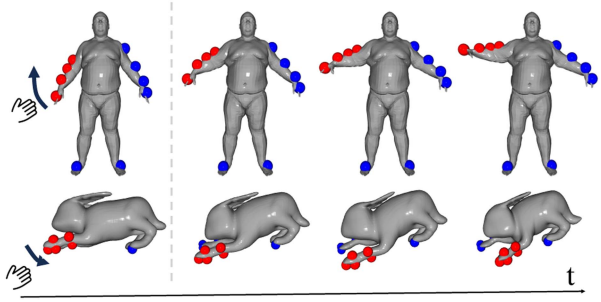


Fig. 15. Shape manipulation by sparse handle movements. Given a 3D shape as input, we edit the shape by dragging a few handles for regions of interest. Red spheres indicate displaced handles, while blue spheres represent fixed constraints. Our model propagates local edits into globally consistent deformations, producing realistic surface motions even for unseen shapes.

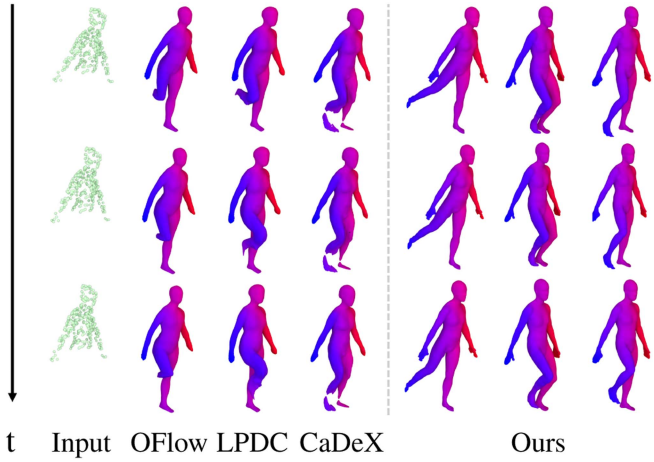


Fig. 16. Qualitative comparisons of 4D shape reconstruction from **highly partial** point cloud sequences, such as half-body scans obtained from the D-FAUST [25] dataset. The colors of the meshes encode the correspondence. Our diffusion-based method produces highly complete human shapes with more favorable motions, offering multiple possible outputs that match the input observations.

on both test subsets in the cross-dataset setting."Test Set 1" and "Test Set 2" correspond to "Unseen Motion" and "Unseen Individual" respectively in the in-dataset evaluation in the Section IV-A. Both subsets are unseen individuals in the cross-dataset evaluation. Notably, the performance gaps are larger than those observed on D-FAUST in Table II, which further demonstrates the superior generalization ability of the proposed 4D latent set representation compared to existing methods.

F. Additional Results

1) Shape Manipulation From Sparse Handle Movements: We further showcase the applicability of our method to user-driven shape editing. Given a source mesh, we randomly select ten sparse vertices as control handles, each assigned a target position to specify the desired deformation. The goal is to propagate these sparse constraints to generate a sequence of globally coherent deforming meshes.

This task only requires motion diffusion. We first obtain the shape latent set by encoding the source mesh using the pre-trained shape auto-encoder. The trajectories of the control handles are then encoded into conditioning embeddings for motion diffusion, following the same architecture described in (4). As illustrated in Fig. 15, our model generates realistic global deformations that faithfully respect sparse user edits, even on unseen identities. The results preserve fine geometric details and exhibit coherent surface displacements, highlighting the effectiveness and potential of our learned deformation priors for interactive non-rigid motion synthesis.

2) Highly Partial Scan Sequences: To assess the robustness of our method to extremely ambiguous data, we set up a challenging experiment on the D-FAUST [25] dataset. This involved reconstructing whole body motions based on partial point clouds of the upper bodies. This setup creates a highly ambiguous scenario, as the same upper body motion can correspond to many different lower-body motions. We adopt the same configuration as Section IV-A, with a frame length ($T = 17$) and input point cloud size ($L = 300$). As shown in Fig. 16, OFlow [15], LPDC [16], and CaDex [17] face challenges in reconstructing the complete shape, often producing distorted shapes such as broken feet. In contrast, our method excels in reconstructing more complete geometries while achieving temporally coherent tracking. Additionally, our approach presents a diverse range of plausible full-body reconstructions that align

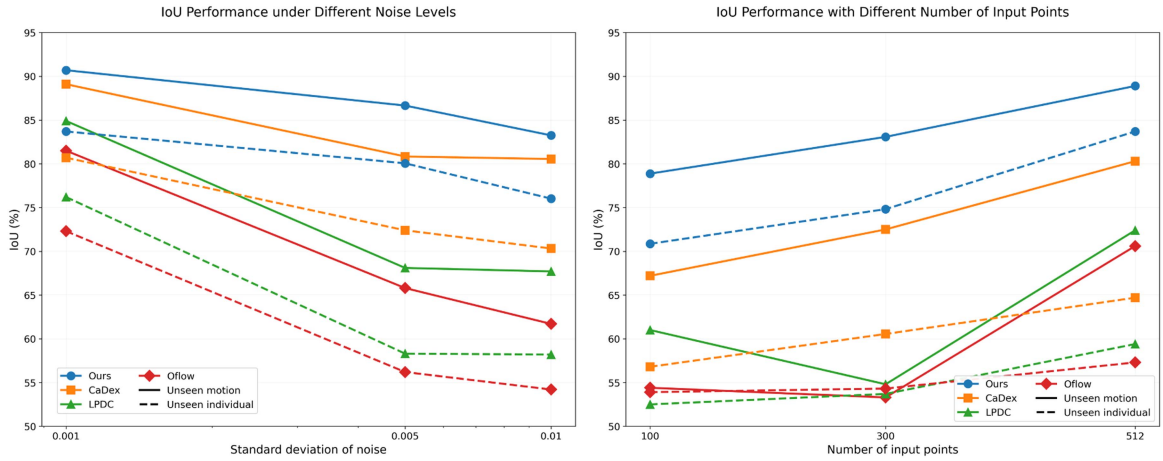


Fig. 17. **Robustness analysis** of Motion2VecSets and previous methods. Left: Results on the D-FAUST [25] dataset under varying noise levels. Right: Results on the DT4D-A [26] dataset with different numbers of input points. Colors and marker shapes denote different methods, while line styles indicate evaluation settings (solid: unseen motions; dashed: unseen individuals).

with the given upper-body scans. The superior performance is primarily attributed to the 4D latent set diffusion. Our diffusion-based method is more capable of tackling the 'one-to-many' complexities from extremely partial data.

3) Long Sequence Reconstruction: Although our method is trained on 4D sequences of fixed length $T = 17$, it can be naturally extended to longer sequences without retraining. We achieve this by splitting long videos into overlapping 17-frame clips and applying our 4D diffusion model autoregressively across time. As shown in Fig. 14, our method generates temporally consistent and structurally coherent deformations over 100-frame sequences from the D-FAUST [25] and DT4D-A [26] datasets, demonstrating strong scalability and robustness for long-range 4D shape generation.

4) Real-World Data Generalization: We validate our model on real-world human scans from the BEHAVE dataset [108], which captures RGB-D data using four Kinect sensors. To align with our partial scan setting, we use a single fixed-view RGB-D stream and back-project the depth map into a partial 3D point cloud as input. Fig. 18 illustrates our reconstruction results alongside the corresponding RGB input. Even in the presence of severe occlusions, such as limbs obscured by objects, our model successfully infers plausible and complete surface reconstructions. This robustness stems from the learned latent distribution and the generative power of the diffusion model, which enables it to reason about missing geometry. Remarkably, our approach maintains high-quality results without any fine-tuning on real data. It also handles sensor noise effectively, demonstrating strong generalization to real-world, imperfect observations.

To demonstrate the generalization and robustness of our model on real-world monocular RGB videos, we use TripoSG [109] to obtain initial reconstructions from each RGB frame, and then sample dynamic point clouds as inputs for our model. The results are visualized in Fig. 19. Our method can still reconstruct plausible geometries and reliable motions with temporal correspondence for animals and articulated objects, such as a bear and a laptop.

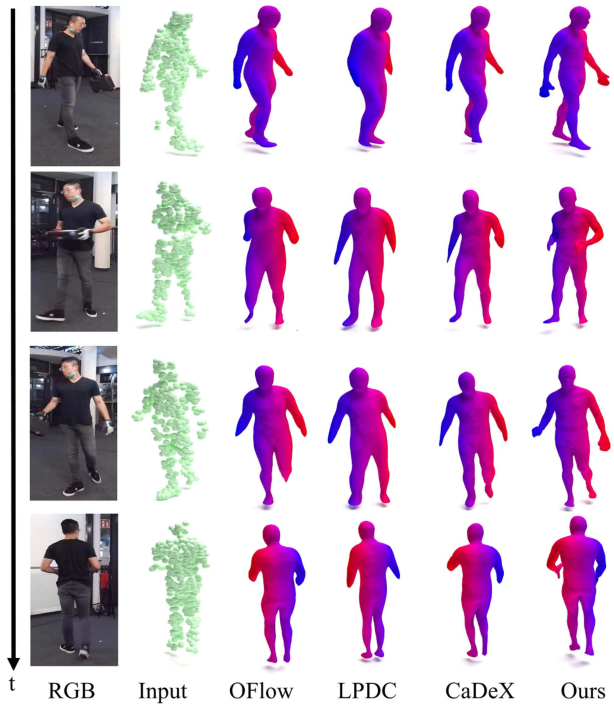


Fig. 18. 4D Shape Completion on the BEHAVE [108] dataset.

5) Failure Cases: While our method shows strong performance for 4D reconstruction, it still has several limitations. We present failure cases in Fig. 20. As shown, our method is inferior at reconstructing geometries of unseen identities that are far from the training data distribution, such as the butterfly and the cloth. It also has difficulty recovering realistic surface deformations that depend on physical properties and simulation, such as the cloth deformations on the right. We expect these issues to be reduced by training on a larger dataset, using stronger data augmentation, and incorporating physical attributes as additional inputs in future work.

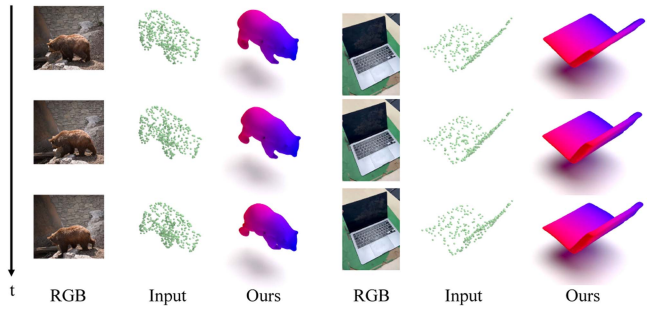


Fig. 19. Real-world generalization results on the sparse point clouds estimated by TripoSG [109] from monocular RGB videos.

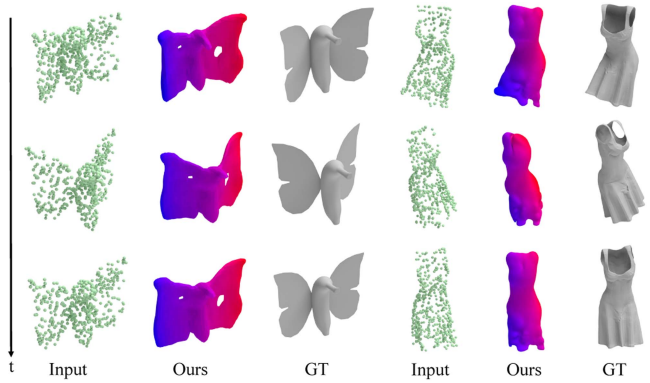


Fig. 20. Failure cases. Our method shows limitations when reconstructing identities that are far from the training data distribution, such as the butterfly and the cloth. It also has difficulty recovering realistic cloth deformations that require physical properties and simulation.

G. Robustness Analysis

We evaluate the robustness of our model on the D-FAUST [25] and DT4D-A [26] datasets under various input degradations. For D-FAUST, we inject Gaussian noise with standard deviations of 0.001, 0.005, and 0.01. For DT4D-A, we vary the number of input points (100, 300, 500). As shown in Fig. 17, our method consistently outperforms state-of-the-art baselines and generalizes better to unseen subjects under various ambiguous observations.

H. Runtime and Memory

In Table IX, we report the computational cost of each method, including inference time, training memory, and the total number of trainable parameters. The runtime evaluation is conducted on the D-FAUST [25] dataset with a sparse setting of $L = 300$ points per frame. We also provide a detailed runtime breakdown for each stage of our pipeline under this configuration. Notably, the inference time remains consistent across different input modalities (e.g., point cloud, voxel grid, image), as the conditional encoders are lightweight relative to the core backbone networks. Due to the multiple denoising steps, our method runs slower than conventional feed-forward methods. The inference speed can be improved by applying distribution matching distillation [110].

TABLE IX
COMPUTATIONAL EFFICIENCY ANALYSIS. **TOP:** COMPARISON WITH STATE-OF-THE-ART METHODS. **BOTTOM:** RUNTIME BREAKDOWN OF OUR M2V PIPELINE.

Method	Avg. Time (s) ↓	Avg. GPU (GB) ↓	Params (M) ↓
OFlow [15]	1.800	0.265	1.2
LPDC [16]	1.769	0.453	8.5
CaDeX [17]	4.357	5.814	2.1
Ours	9.479	4.358	5.3

Stage (Ours)	Time (s) ↓	Percentage (%)
Shape Diffusion	3.595	37.9%
Deformation Diffusion	5.384	56.8%
Mesh Generation	0.352	3.7%
Total	**9.479**	**100.0%**

V. CONCLUSION, LIMITATIONS, AND FUTURE WORK

We presented Motion2VecSets, a 4D diffusion model for dynamic surface reconstruction and generation from various imperfect inputs, including sparse point clouds, coarse voxel grids, and RGB images. To enable high-quality 4D diffusion, we introduced a 4D latent set representation that compactly encodes initial geometry and pairwise deformations using transformer-based encoders and decoders. Our model captures the probabilistic distribution of non-rigid shape and motion through iterative denoising, enabling plausible and diverse outputs. To ensure temporally smooth motion, we jointly diffuse stacked deformation latent sets across frames. For computational efficiency, we design an interleaved space-time attention block that synchronizes deformation latents over time while significantly reducing memory usage. Unlike global latent representations, our 4D latent set offers more accurate modeling of complex nonlinear motions and improves generalization to unseen identities and motion patterns. Extensive experiments across multiple datasets, including D-FAUST (humans), DT4D-A (animals), and S2M (articulated objects), demonstrate the effectiveness of our approach in reconstructing and tracking non-rigid objects from ambiguous inputs.

While our method achieves notable progress in 4D reconstruction, it has several limitations. First, it relies on direct 4D supervision from dynamic mesh sequences, which are difficult to collect at large scale. As a result, more diverse and larger datasets are still needed to learn more general 4D shape priors. A promising direction is to explore hybrid supervision combining mesh and video data to improve generalization. Second, our current framework does not incorporate texture reconstruction or generation, which is important for high-fidelity applications. Third, our framework focuses on single-object modeling; extending it to dynamic scenes with multiple interacting objects remains an open challenge. Finally, our deformation priors are learned purely from data, without incorporating physical constraints. Integrating physics-based priors could yield more realistic and physically plausible motion patterns for objects such as hair, cloth, and fluids. We leave these challenges for future work.

REFERENCES

[1] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," in *Seminal Graphics: Pioneering Efforts That Shaped the Field*. New York, NY, USA: ACM, 1998, pp. 347–353.

[2] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Proc. 4th Eurographics Symp. Geometry Process.*, 2006, pp. 61–70.

[3] R. A. Newcombe et al., "KinectFusion: Real-time dense surface mapping and tracking," in *Proc. 10th IEEE Int. Symp. Mixed Augmented Reality*, 2011, pp. 127–136.

[4] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2006, pp. 519–528.

[5] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 343–352.

[6] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "VolumeDeform: Real-time volumetric non-rigid reconstruction," in *Proc. 14th Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, 2016, pp. 362–379.

[7] T. Yu et al., "DoubleFusion: Real-time capture of human performances with inner body shapes from a single depth sensor," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7287–7296.

[8] O. Sorkine and M. Alexa, "As-rigid-as-possible surface modeling," in *Proc. Symp. Geometry Process.*, 2007, pp. 109–116.

[9] R. W. Sumner, J. Schmid, and M. Pauly, "Embedded deformation for shape manipulation," *ACM Trans. Graph.*, vol. 26, pp. 80–es, 2007.

[10] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A skinned multi-person linear model," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 248:1–248:16, Oct. 2015.

[11] A. A. A. Osman, T. Bolkart, and M. J. Black, "STAR: A sparse trained articulated human body regressor," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 598–613. [Online]. Available: https://star.is.tue.mpg.de

[12] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," *ACM Trans. Graph.*, vol. 36, no. 6, Nov. 2017, Art. no. 245.

[13] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero, "Learning a model of facial shape and expression from 4D scans," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 194:1–194:17, 2017, doi: 10.1145/3130800.3130813.

[14] S. Zuffi, A. Kanazawa, D. Jacobs, and M. J. Black, "3D menagerie: Modeling the 3D shape and pose of animals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5524–5532.

[15] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Occupancy flow: 4D reconstruction by learning particle dynamics," in *Proc. Int. Conf. Comput. Vis.*, Oct. 2019, pp. 5378–5388.

[16] J. Tang, D. Xu, K. Jia, and L. Zhang, "Learning parallel dense correspondence from spatio-temporal descriptors for efficient and robust 4D reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 6022–6031.

[17] J. Lei and K. Daniilidis, "CaDeX: Learning canonical deformation coordinate space for dynamic surface representation via neural homeomorphism," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 6614–6624. [Online]. Available: https://cis.upenn.edu/leijh/projects/cadex

[18] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3D reconstruction in function space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4455–4465.

[19] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 165–174.

[20] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3D reconstruction in function space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4460–4470.

[21] S. Peng, M. Niemeyer, L. Mescheder, M. Pollefeys, and A. Geiger, "Convolutional occupancy networks," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 523–540.

[22] B. Zhang, J. Tang, M. Nießner, and P. Wonka, "3DShape2VecSet: A 3D shape representation for neural fields and generative diffusion models," *ACM Trans. Graph.*, vol. 42, no. 4, Jul. 2023, Art. no. 92. doi: 10.1145/3592442.

[23] X. Zhang, N. Li, and A. Dai, "DNF: Unconditional 4D generation with dictionary-based neural fields," in *Proc. Comput. Vis. Pattern Recognit. Conf.*, 2025, pp. 26047–260 56.

[24] W. Cao, C. Luo, B. Zhang, M. Nießner, and J. Tang, "Motion2VecSets: 4D latent vector set diffusion for non-rigid shape reconstruction and tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 20496–20506.

[25] F. Bogo, J. Romero, G. Pons-Moll, and M. J. Black, "Dynamic FAUST: Registering human bodies in motion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5573–5582.

[26] Y. Li, H. Takehara, T. Taketomi, B. Zheng, and M. Nießner, "4DComplete: Non-rigid motion estimation beyond the observable surface," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 12686–12696.

[27] J. Mu, W. Qiu, A. Kortylewski, A. Yuille, N. Vasconcelos, and X. Wang, "A-SDF: Learning disentangled signed distance functions for articulated shape representation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 13001–13011.

[28] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3D-R2N2: A unified approach for single and multi-view 3D object reconstruction," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 628–644.

[29] X. Han, Z. Li, H. Huang, E. Kalogerakis, and Y. Yu, "High-resolution shape completion using deep neural networks for global structure and local geometry inference," in *Proc. IEEE Int. Conf. Comput. Vis.*. Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2017, pp. 85–93. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICCV.2017.19

[30] M. Gadelha, S. Maji, and R. Wang, "3D shape induction from 2D views of multiple objects," in *Proc. Int. Conf. 3D Vis.*, 2017, pp. 402–411.

[31] D. Stutz and A. Geiger, "Learning 3D shape completion from laser scan data with weak supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1955–1964.

[32] H. Fan, H. Su, and L. J. Guibas, "A point set generation network for 3D object reconstruction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 605–613.

[33] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3D point clouds," in *Proc. Int. Conf. Mach. Learn. PMLR*, 2018, pp. 40–49.

[34] H. Zhao, L. Jiang, J. Jia, P. H. Torr, and V. Koltun, "Point transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 16259–16 268.

[35] G. Riegler, A. O. Ulusoy, and A. Geiger, "OctNet: Learning deep 3D representations at high resolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3577–3586.

[36] P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, and X. Tong, "O-CNN: Octree-based convolutional neural networks for 3D shape analysis," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–11, 2017.

[37] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Octree generating networks: Efficient convolutional architectures for high-resolution 3D outputs," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2107–2115.

[38] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2Mesh: Generating 3D mesh models from single RGB images," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 52–67.

[39] T. Groueix, M. Fisher, V. G. Kim, B. Russell, and M. Aubry, "AtlasNet: A Papier-mâché approach to learning 3D surface generation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 216–224.

[40] Y. Liao, S. Donné, and A. Geiger, "Deep marching cubes: Learning explicit surface representations," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2916–2925.

[41] J. Tang, X. Han, J. Pan, K. Jia, and X. Tong, "A skeleton-bridged deep learning approach for generating meshes of complex topologies from single RGB images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4536–4545.

[42] J. Pan, X. Han, W. Chen, J. Tang, and K. Jia, "Deep mesh reconstruction from single RGB images via topology modification networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9963–9972.

[43] J. Tang, X. Han, M. Tan, X. Tong, and K. Jia, "SkeletonNet: A topology-preserving solution for learning mesh reconstruction of object surfaces from RGB images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 6454–6471, Oct. 2022.

[44] Z. Chen and H. Zhang, "Learning implicit fields for generative shape modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5939–5948.

[45] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 165–174.

[46] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman, "Implicit geometric regularization for learning shapes," in *Proc. Mach. Learn. Syst.* 2020, pp. 3569–3579.

[47] J. Chibane, T. Alldieck, and G. Pons-Moll, "Implicit functions in feature space for 3D shape reconstruction and completion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6970–6981.

[48] R. Chabra et al., "Deep local shapes: Learning local SDF priors for detailed 3D reconstruction," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 608–625.

[49] E. Tretschk, A. Tewari, V. Golyanik, M. Zollhöfer, C. Stoll, and C. Theobalt, "PatchNets: Patch-based generalizable deep implicit 3D shape representations," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 293–309.

[50] C. Chen, Y.-S. Liu, and Z. Han, "GridPull: Towards scalability in learning implicit representations from 3D point clouds," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 18276–18288.

[51] J. Tang, J. Lei, D. Xu, F. Ma, K. Jia, and L. Zhang, "SA-ConvONet: Sign-agnostic optimization of convolutional occupancy networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 6504–6513.

[52] J. Tang, L. Markhasin, B. Wang, J. Thies, and M. Nießner, "Neural shape deformation priors," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 17117–17132.

[53] P. Palafox, A. Božič, J. Thies, M. Nießner, and A. Dai, "NPMs: Neural parametric models for 3D deformable shapes," 2021, *arXiv:2104.00702*.

[54] B. Jiang, Y. Zhang, X. Wei, X. Xue, and Y. Fu, "Learning compositional representation for 4D captures with neural ODE," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 5336–5346.

[55] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 6840–6851.

[56] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," 2020, *arXiv:2011.13456*.

[57] A. Nichol et al., "GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models," 2021, *arXiv:2112.10741*.

[58] P. Dhariwal and A. Nichol, "Diffusion models beat GANs on image synthesis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 8780–8794.

[59] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," 2022,*arXiv:2204.06125*.

[60] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 10684–10 695.

[61] A. Blattmann, R. Rombach, K. Oktay, J. Müller, and B. Ommer, "Retrieval-augmented diffusion models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 15309–15324.

[62] A. Blattmann et al., "Stable video diffusion: Scaling latent video diffusion models to large datasets," 2023, *arXiv:2311.15127*.

[63] Y. Guo et al., "AnimateDiff: Animate your personalized text-to-image diffusion models without specific tuning," 2023, *arXiv:2307.04725*.

[64] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 8633–8646.

[65] H. Liu et al., "AudioLDM: Text-to-audio generation with latent diffusion models," 2023, *arXiv:2301.12503*.

[66] D. Yang et al., "Diffsound: Discrete diffusion model for text-to-sound generation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 31, pp. 1720–1733, 2023.

[67] X. Li, J. Thickstun, I. Gulrajani, P. S. Liang, and T. B. Hashimoto, "Diffusion-LM improves controllable text generation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 4328–4343.

[68] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. Van Den Berg, "Structured denoising diffusion models in discrete state-spaces," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 17981–17993.

[69] S. Gong, M. Li, J. Feng, Z. Wu, and L. Kong, "DiffuSeq: Sequence to sequence text generation with diffusion models," 2022, *arXiv:2210.08933*.

[70] J. Ho et al., "Imagen video: High definition video generation with diffusion models," 2022, *arxiv:2210.02303*.

[71] S. Luo and W. Hu, "Diffusion probabilistic models for 3D point cloud generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*,2021, pp. 2836–2844.

[72] L. Zhou, Y. Du, and J. Wu, "3D shape generation and completion through point-voxel diffusion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 5826–5835.

[73] G. Chou, Y. Bahat, and F. Heide, "Diffusion-SDF: Conditional generative modeling of signed distance functions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 2262–2272.

[74] J. Tang, Y. Nie, L. Markhasin, A. Dai, J. Thies, and M. Nießner, "DiffuScene: Denoising diffusion models for generative indoor scene synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 20507–20518.

[75] Y. Siddiqui, J. Thies, F. Ma, Q. Shan, M. Nießner, and A. Dai, "Texturify: Generating textures on 3D shape surfaces," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 72–88.

[76] Y. Zhang et al., "DreamMat: High-quality PBR material generation with geometry-and light-aware diffusion models," *ACM Trans. Graph.*, vol. 43, no. 4, pp. 1–18, 2024.

[77] D. Z. Chen, Y. Siddiqui, H.-Y. Lee, S. Tulyakov, and M. Nießner, "Text2Tex: Text-driven texture synthesis via diffusion models," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 18558–18568.

[78] R. Jiang et al., "AvatarCraft: Transforming text into neural human avatars with parameterized shape and pose control," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 14371–14382.

[79] N. Kolotouros, T. Alldieck, A. Zanfir, E. Bazavan, M. Fieraru, and C. Sminchisescu, "DreamHuman: Animatable 3D avatars from text," in *Proc. Adv. Neural Inf. Process. Syst.*, 2023, pp. 10516–10529.

[80] J. Tang, A. Dai, Y. Nie, L. Markhasin, J. Thies, and M. Nießner, "DPHMs: Diffusion parametric head models for depth-based tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 1111–1122.

[81] B. L. Bhatnagar, X. Xie, I. A. Petrov, C. Sminchisescu, C. Theobalt, and G. Pons-Moll, "BEHAVE: Dataset and method for tracking human object interactions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 15935–15946.

[82] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "GRAB: A dataset of whole-body human grasping of objects," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 581–600.

[83] Y. Huang, O. Taheri, M. J. Black, and D. Tzionas, "InterCap: Joint markerless 3D tracking of humans and objects in interaction," in *Proc. DAGM German Conf. Pattern Recognit.*, 2022, pp. 281–299.

[84] M. Hassan, P. Ghosh, J. Tesch, D. Tzionas, and M. J. Black, "Populating 3D scenes by learning human-scene interaction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14708–14718.

[85] A. Vahdat et al., "LION: Latent point diffusion models for 3D shape generation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 10021–10039.

[86] B. Mildenhall, P.P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," *Commun. ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[87] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D Gaussian splatting for real-time radiance field rendering," *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[88] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "DreamFusion: Text-to-3D using 2D diffusion," 2022, *arXiv:2209.14988*.

[89] J. R. Shue, E. R. Chan, R. Po, Z. Ankner, J. Wu, and G. Wetzstein, "3D neural field generation using triplane diffusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 20875–20886.

[90] B. Zhang and P. Wonka, "LaGeM: A large geometry model for 3D representation learning and diffusion," 2024, *arXiv:2410.01295*.

[91] J. Xiang et al., "Structured 3D latents for scalable and versatile 3D generation," in *Proc. Comput. Vis. Pattern Recognit. Conf.*, 2025, pp. 21469–21480.

[92] S. Wu et al., "Direct3D: Scalable image-to-3D generation via 3D latent diffusion transformer," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 121 859–121 881.

[93] X. Yang et al., "Hunyuan3D 1.0: A unified framework for text-to-3D and image-to-3D generation," 2024, *arXiv:2411.02293*.

[94] Z. Zhao et al., "Hunyuan3D 2.0: Scaling diffusion models for high resolution textured 3D assets generation," 2025, *arXiv:2501.12202*.

[95] J. Lei, C. Deng, B. Shen, L. J. Guibas, and K. Daniilidis, "NAP: Neural 3d articulated object prior," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 31 878–31 894.

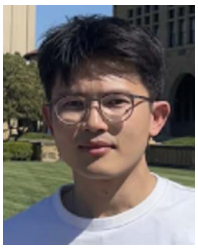
[96] I. Liu et al., "Riganything: Template-free autoregressive rigging for diverse 3D assets," *ACM Trans. Graph. (TOG)*, vol. 44, no. 4, pp. 1–12, 2025.

[97] C. Song et al., "MagicArticulate: Make your 3D models articulation-ready," in *Proc. Comput. Vis. Pattern Recognit. Conf.*, 2025, pp. 15998–1600 7.

[98] G. Tevet, S. Raab, B. Gordon, Y. Shafir, D. Cohen-Or, and A. H. Bermano, "Human motion diffusion model," 2022, *arXiv:2209.14916*.

[99] Y. Yuan, J. Song, U. Iqbal, A. Vahdat, and J. Kautz, "PhysDiff: Physics-guided human motion diffusion model," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 16010–16021.
[100] M. Zhang et al., "MotionDiffuse: Text-driven human motion generation with diffusion model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 6, pp. 4115–4128, Jun. 2024.
[101] J. Ren et al., "L4GM: Large 4D gaussian reconstruction model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, pp. 56828–56858.
[102] H. Liang et al., "Diffusion4D: Fast spatial-temporal consistent 4D generation via video diffusion models," 2024, *arXiv:2405.16645*.
[103] Y. Jiang, L. Zhang, J. Gao, W. Hu, and Y. Yao, "Consistent4D: Consistent 360° dynamic object generation from monocular video," 2023, *arXiv:2311.02848*.
[104] T. Karras, M. Aittala, T. Aila, and S. Laine, "Elucidating the design space of diffusion-based generative models," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2022, pp. 26565–26577.
[105] M. Oquab et al., "Dinov2: Learning robust visual features without supervision," 2023, *arXiv:2304.07193*.
[106] X. Wang, B. Zhou, Y. Shi, X. Chen, Q. Zhao, and K. Xu, "Shape2motion: Joint analysis of motion parts and attributes from 3D shapes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8876–8884.
[107] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "Neural ordinary differential equations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 6572–6583.
[108] B. L. Bhatnagar, X. Xie, I. A. Petrov, C. Sminchisescu, C. Theobalt, and G. Pons-Moll, "BEHAVE: Dataset and method for tracking human object interactions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 15914–15925.
[109] Y. Li et al., "TripoSG: High-fidelity 3D shape synthesis using large-scale rectified flow models," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2025.
[110] T. Yin et al., "One-step diffusion with distribution matching distillation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 6613–6623.

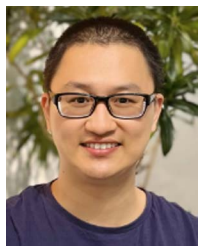
Jiapeng Tang received the BE and MSc degrees from the School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China, in 2014 and 2018, respectively. He is currently working toward the PhD degree with the Department of Informatics, the Technical University of Munich. His research interests include computer vision and graphics, generative models, and deep learning, with a focus on 3D/4D reconstruction and generation.



Wei Cao received the BSc and MSc degrees from the School of Computation, Information and Technology, Technical University of Munich. He is currently working toward the PhD degree with the School of Information Sciences, the University of Illinois Urbana-Champaign. . His research interests include 3D/4D computer vision and autonomous driving.

Biao Zhang received BS and MS degrees from Xi'an Jiaotong University, and the PhD degree from King Abdullah University of Science and Technology (KAUST). He is currently a postdoctoral researcher with KAUST. He has authored or coauthored in conferences including SIGGRAPH, CVPR, ICCV, ICLR, and NeurIPS. His work explores connections between computer graphics and machine learning, with recent focus on generative models and representation learning.

Chang Luo received the master degree in informatics from the Technical University of Munich. He is currently working toward the PhD degree with the Department of Creative Informatics, Graduate School of Information, Science and Technology, the University of Tokyo. His research interest include geometry processing and inverse rendering for computer graphics.



Yaoyao Liu (Member, IEEE) recieved the BS degree in electronic information engineering from Tianjin University, and PhD degree in computer science from the Max Planck Institute for Informatics. He is currently an assistant professor with the School of Information Sciences and the Coordinated Science Laboratory, the University of Illinois Urbana-Champaign. He is also affiliated with the Siebel School of Computing and Data Science, the National Center for Supercomputing Applications, and the Illinois Informatics. His research interests include continual learning, few-shot learning, semi-supervised learning, generative models, 3D geometry models, and medical imaging. He is a recipient of the 2024 ECVA PhD Award.

Matthias Nießner is a professor with the Technical University of Munich, where he leads the Visual Computing Lab. He was a visiting assistant professor with Stanford University. He has authored or coauthored more than 150 academic publications, including 25 papers at the prestigious ACM Transactions on Graphics (SIGGRAPH / SIGGRAPH Asia) journal and 55 works at the leading vision conferences (CVPR, ECCV, ICCV). His research interests include the intersection of computer vision, graphics, and machine learning, where he is particularly interested in cutting-edge techniques for 3D reconstruction, semantic 3D scene understanding, video editing, and AI-driven video synthesis. He has won best paper awards, including at SIGCHI'14, HPG'15, SPG'18, and the SIGGRAPH'16 Emerging Technologies Award for the best Live Demo, for his several publications.