






Preconditioned Deformation Grids

Julian Kaltheuner¹ , Alexander Oebel¹ , Hannah Droege¹ , Patrick Stotko¹ , Reinhard Klein¹ 

¹University of Bonn, Germany

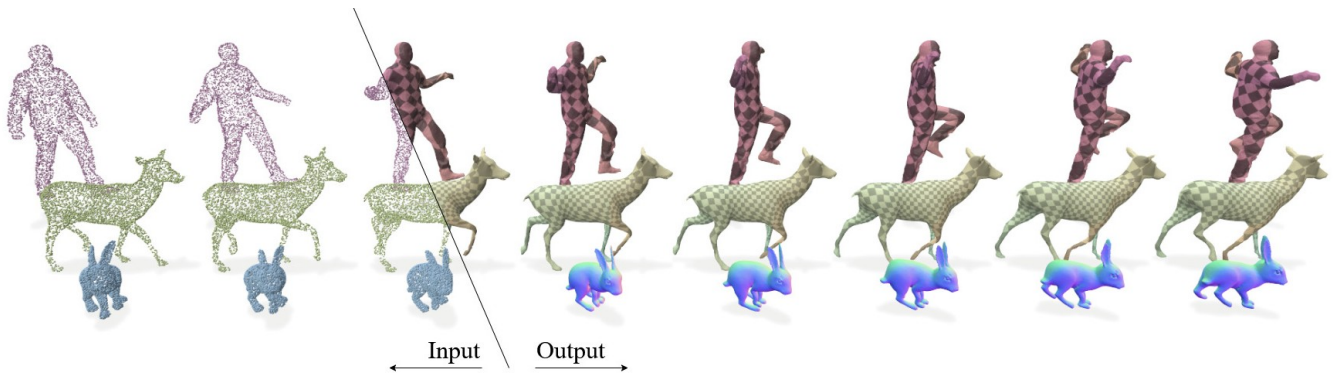


Figure 1: Raw point cloud sequences (left) are unstructured and lack temporal correspondences. We propose Preconditioned Deformation Grids (right), a method for temporally coherent, high-fidelity surface reconstructions that evolve smoothly over time.

Abstract

Dynamic surface reconstruction of objects from point cloud sequences is a challenging field in computer graphics. Existing approaches either require multiple regularization terms or extensive training data which, however, lead to compromises in reconstruction accuracy as well as over-smoothing or poor generalization to unseen objects and motions. To address these limitations, we introduce Preconditioned Deformation Grids, a novel technique for estimating coherent deformation fields directly from unstructured point cloud sequences without requiring or forming explicit correspondences. Key to our approach is the use of multi-resolution voxel grids that capture the overall motion at varying spatial scales, enabling a more flexible deformation representation. In conjunction with incorporating grid-based Sobolev preconditioning into gradient-based optimization, we show that applying a Chamfer loss between the input point clouds as well as to an evolving template mesh is sufficient to obtain accurate deformations. To ensure temporal consistency along the object surface, we include a weak isometry loss on mesh edges which complements the main objective without constraining deformation fidelity. Extensive evaluations demonstrate that our method achieves superior results, particularly for long sequences, compared to state-of-the-art techniques.

CCS Concepts

• Computing methodologies → Reconstruction; Mesh models;

1. Introduction

Reconstructing dynamic 3D surfaces from temporal point cloud sequences is a fundamental problem in computer graphics with diverse applications in animation, virtual production, medical imaging, robotics, autonomous driving, as well as augmented reality (AR) and virtual reality (VR). The increasing availability of commodity depth sensing, ranging from smartphone LiDAR to multi-camera RGB systems, has made it feasible to capture dense 3D point clouds at video rates. However, these point clouds are typ-

ically unstructured, lack temporal correspondences, and exhibit noise or incompleteness, posing significant challenges for reliable surface reconstruction. Naively applying static reconstruction methods [HGA*23, PJJ*21, HMGCO20, WSS*19] to each frame independently fails to exploit temporal coherence, often resulting in inconsistent geometry, lost correspondences, and high computational cost. Thus, dynamic reconstruction methods aim to estimate a consistent 3D surface for a reference frame and deform it to match subsequent points measurements across time. Yet, this problem is

inherently under-constrained, and in the absence of strong temporal cues, deformation fields may overfit the data while producing implausible motion.

To mitigate these issues, previous approaches incorporate temporal priors to enforce continuity across frames. Template-based models such as SMPL [LMR*15] and SCAPE [ASK*05] encode correspondences via parametric shape spaces, while alternative methods estimate inter-frame deformations through learning-based or optimization-driven techniques [BPZ*21, LD22]. Although effective in specific domains, these approaches often rely on category-specific assumptions or extensive training data, limiting their ability to generalize to unseen object classes or non-rigid, unpredictable motions. In addition, many methods employ strong regularization to stabilize reconstructions and enforce smoothness which often comes at the cost of geometric detail [WLLY19, YGL*19]. This trade-off between fidelity and stability can undermine subtle but meaningful surface variations, motivating the need for more flexible formulations. To overcome these limitations, it is essential to develop a framework that jointly exploits temporal coherence without relying on restrictive priors, while still preserving high-frequency details across challenging dynamic sequences.

In this work, we introduce *Preconditioned Deformation Grids*, a correspondence-free, training-free framework for reconstructing temporally coherent, high-fidelity surfaces directly from unstructured point cloud sequences. Our method begins by selecting a suitable keyframe to estimate an initial deformable template mesh, which is further refined as part of the optimization process. Rather than relying on explicit correspondences or a set of carefully hand-tuned regularization terms, our approach employs Sobolev preconditioning, which spatially diffuses the under-constrained gradient information from the raw point clouds across local neighborhoods. This effectively acts as a spatially adaptive smoothness constraint, allowing the optimization to prioritize plausible deformations without explicitly imposing any restrictions to the reconstruction fidelity. To further guide the process, we represent the overall motion at various spatial scales using multi-resolution voxel grids. Coarser grid levels represent broad movements and thus help maintain temporal coherence over long sequences, while finer levels enable the recovery of high-frequency surface details by allowing localized adjustments. Thanks to the stabilizing effect of Sobolev preconditioning and the multi-scale representation of the deformation field, a simple Chamfer loss defined on the unstructured input point cloud together with a weak isometry loss on the edges of the evolving template mesh is sufficient to achieve superior results over current state-of-the-art approaches.

In summary, our main contributions are:

- A correspondence-free deformation framework that operates directly on unstructured point clouds, eliminating the need for predefined template models or category-specific priors, and enables robust, large-scale motion estimation for arbitrary objects.
- A multi-resolution voxel grid representation for the deformation field that models the motion at varying scales.
- A grid-based Sobolev preconditioning scheme that stabilizes the optimization by diffusing under-constrained gradients across neighboring grid cells.

The code of our work is available at <https://github.com/vc-bonn/preconditioned-deformation-grids>.

2. Related Work

2.1. Parametric Template Models

A widely adopted strategy for category-specific 3D reconstruction involves deforming a predefined mesh template to align with observed data. In facial reconstruction, the FLAME [LBB*17] and FaceVerse [WCY*22] models define parametric spaces over shape and expression. For full-body reconstruction, approaches like SMPL [LMR*15], SMPL-X [PCG*19], SCAPE [ASK*05], as well as the more recent GHUM/GHUML [XBZ*20] similarly encode pose and identity within low-dimensional manifolds and use corrective blend shapes and linear blend skinning to deform the template. These models have been successfully applied to detailed human performance capture from real-world recordings [BKL*16, SBFB19, LHR*21], and further extended to incorporate clothing and apparel variations [BTTPM19, AMB*19, BTTPM20].

While template-based approaches offer compact, interpretable control over shape and articulation, they are fundamentally limited by the expressiveness of the underlying parametric model, which can restrict generalization to novel shapes, poses, or deformable objects outside the training distribution.

2.2. Deformation Field Estimation

Estimating non-rigid motion from image or point cloud data is commonly addressed through either optimization-based or learning-based methods that solve for deformation fields. Early optimization-driven techniques model deformation directly from input data. Deformation graphs, introduced by Sumner et al. [SSP07], provide a sparse and flexible representation for surface motion. This concept was extended in DynamicFusion [NFS15], which estimates a volumetric 6D warp field to incrementally align a canonical surface with each new input frame. Recently, the success of 3D Gaussian Splatting (3DGS) [KKLD23] in efficiently representing and rendering radiance fields led to further advances in dynamic reconstruction. For instance, 4DTAM [MBD25] performed online 4D tracking and mapping from a single RGB-D stream using dynamic surface Gaussians, jointly optimizing geometry, appearance, camera ego-motion, and a learned warp field. Other Gaussian-Splatting-based dynamic SLAM systems distinguished between static and dynamic scene content, either by separating them into individual Gaussian maps [LSS*25] or by predicting uncertainty images for guiding rigid bundle adjustment [ZZB*25].

More recently, learning-based methods have gained traction. Neural Deformation Graphs [BPZ*21] and Neural Non-Rigid Tracking [BPZ*20] learn non-rigid motion by estimating point correspondences and deformation fields through neural networks. Hierarchical models such as Neural Deformation Pyramid [LH22] represent motion at multiple spatial scales, where each level is handled by a lightweight MLP. An alternative line of work formulates deformation as a learned, continuous spatio-temporal vector field [NMOG19], integrated via Neural ODE solvers [CRBD18], or using them to update latent codes that control global shapes

[JZW*21] or for mesh deformations [HJL*20], whereas other approaches [TXJZ21] predict deformations directly. Complementary to these vector-field formulations, Dynamic Neural Surfaces [NLW*25] represent deforming 4D shapes as continuous elastic surfaces in space–time, enabling spatio-temporal registration and statistical analysis for genus-zero surfaces. Recent generative approaches further explore probabilistic modeling of deformation. Motion2VecSets [CLZ*24] employs a 4D diffusion model to learn distributions over shapes and their deformations. Other methods aim to jointly learn a canonical shape and its deformation mappings, either via latent embeddings [LD22, YTB*21] or within volumetric rendering frameworks [PCPMN21, PSB*21, LNSW21], where deformation is optimized alongside radiance fields. In contrast to models that rely on shape priors, DynoSurf [YRH*24] introduces an unsupervised method that jointly estimates a deforming surface and template geometry directly from point cloud sequences.

In the same spirit, our method operates in a category-agnostic, training-free regime, estimating deformation fields without relying on strong priors, supervision, or correspondence annotations.

2.3. Preconditioning

Preconditioning is a classical strategy in inverse problems used to accelerate convergence and enhance optimization stability by carefully adapting update steps. In geometry processing, preconditioners have been employed to improve mesh parameterization [CBSS17] and accelerate mesh deformation [KGL16]. Krishna et al. [KFS13] demonstrate the effectiveness of multi-level preconditioning on Laplacian matrices, showing broad applicability across mesh-based tasks. Despite these advances, ill-conditioned optimization problems remain a challenge in high-dimensional and irregular domains. Recent works explore learning-based alternatives, using graph neural networks to learn effective preconditioners from data [RFM*24, LCDM23, Che25, HÖS24, TRI*24].

A particularly relevant technique in this context is Sobolev preconditioning, which replaces the standard L^2 inner product with a Sobolev (e.g., H^1) inner product to compute smoother gradient directions [Neu85]. Sobolev gradient methods have been widely studied for applications in surface smoothing and minimal surface flows [Ren04, RN95, EPT*07]. Martin et al. [MJBC13] further demonstrated the benefits of Sobolev preconditioners in the area of geometry processing and optimized shapes for smooth surfaces. The technique has since been applied to non-rigid scene reconstruction from RGB-D data [SBI18], differentiable rendering pipelines [NJJ21], and registration of deforming objects [JKYL25]. Recently, Chang et al. [CYB*24] proposed a variant that uses spatiotemporal bilateral gradient filtering, which diffuses gradient information for stability while preserving high-frequency details.

In this work, we adopt a grid-based Sobolev preconditioning scheme to regularize gradient updates from unstructured point cloud sequences, promoting smooth yet flexible deformation fields that improve spatial coherence without sacrificing geometric detail.

3. Method

We present an optimization-based method for estimating dense surface deformations from unstructured point cloud sequences, with-

out relying on temporal correspondences, learned priors, or pre-training. Given a sequence of point clouds $\{\mathcal{P}_t\}_{t=0}^T$, where each $\mathcal{P}_t = \{\mathbf{p}_{i,t} \in \mathbb{R}^3\}$, and an initial mesh $\mathcal{X}_0 = \{\mathbf{x}_{i,0} \in \mathbb{R}^3\}$ defined at a keyframe (see Sec. 3.5), our goal is to estimate a sequence of deformation fields that, when applied to \mathcal{X}_0 , yield temporally consistent surface reconstructions $\{\hat{\mathcal{X}}_t\}_{t=1}^T$. As illustrated in Fig. 2, we represent the deformation field as a multi-resolution voxel grid which encodes local rigid transformations at different spatial scales (see Sec. 3.1). A key technical contribution is a spatially-aware preconditioning scheme (see Sec. 3.2) that stabilizes the optimization by enforcing spatial smoothness in the deformation field.

3.1. Multi-Resolution Transformation Grid

To enable the estimation of temporally coherent deformations, we represent the motion from time step $t - 1$ to t at different spatial scales using a multi-resolution voxel grid $\mathcal{G}_t = \{\mathcal{G}_t^l\}_{l=1}^L$ composed of L hierarchical levels. Each level \mathcal{G}_t^l consists of a finite set of grid cells $\mathcal{C}^l \subseteq \{c \in [0 : 2l - 1]^3\}$, where each cell c stores a 6D transformation vector $\mathbf{T}_t^{l,c} \in \mathbb{R}^6$, parameterizing a local rigid motion via rotation and translation (see Sec. 3.3). In particular, the spatial resolution of the grid increases linearly by $2l - 1$ with level index l , ranging a coarse global deformation at $l = 1$ to fine-scale local displacements at the finest level $L = 10$. For an input point $\mathbf{x} \in \mathbb{R}^3$, the corresponding transformation $\hat{\mathbf{T}}_t(\mathbf{x}) \in \mathbb{R}^6$ is computed by aggregating the trilinear-interpolated partial transformations $\mathbf{T}_t^{l,c}$ from neighboring cells $c \in \mathcal{N}_t^l(\mathbf{x})$ across all resolution levels l :

$$\hat{\mathbf{T}}_t(\mathbf{x}) = \frac{1}{L} \sum_{l=1}^L \sum_{c \in \mathcal{N}_t^l(\mathbf{x})} w_t^{l,c}(\mathbf{x}) \cdot \mathbf{T}_t^{l,c}, \quad (1)$$

where $w_t^{l,c}$ are the trilinear interpolation weights. Our approach constructs the deformation field using a redundant, multi-scale representation, achieved by explicitly averaging transformations derived from parameters at each resolution level. This design allows parameters at each level to model aspects of the *absolute deformation* relevant to their respective scale, rather than encoding explicit *residuals* of coarser levels. The inherent redundancy in an averaged representation is crucial for enhancing optimization stability, as it provides robustness against the high-frequency variations and estimation sensitivities often encountered when learning direct residuals, which can be challenging even for preconditioned solvers.

3.2. Spatial Smoothness via Grid Preconditioning

Real-world deformations exhibit strong spatial coherence: neighboring regions typically undergo similar transformations. In the absence of regularization, the deformation estimation problem is severely under-constrained since the input points \mathcal{P}_t provide only sparse observations, allowing many plausible interpolated deformations between observed points. Naively optimizing transformations for each voxel independently can result in folding, tearing, or discontinuous motion fields that violate physical plausibility.

To address this, rather than imposing spatial coherence through an explicit penalty term, we incorporate smoothness directly into the optimization dynamics via preconditioning. Let \mathbf{T}^l denote the

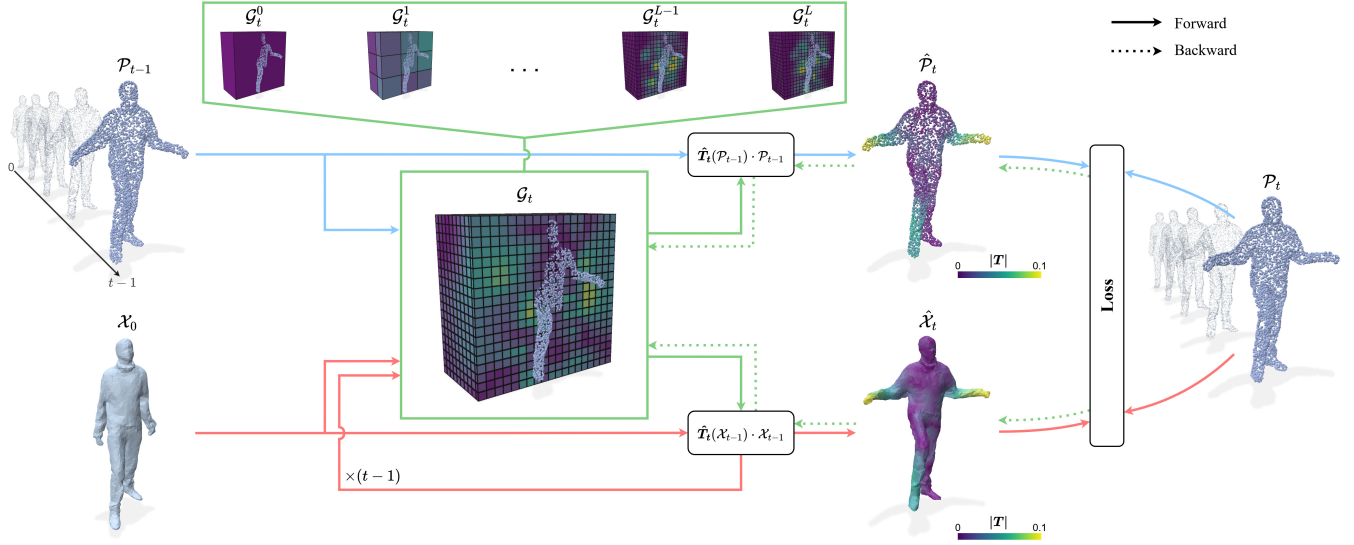


Figure 2: Overview of our correspondence-free deformation framework. At the heart of our approach is a multi-resolution voxel grid \mathcal{G}_t that encodes 6D transformations at multiple spatial scales l . Given a sequence of unstructured point clouds \mathcal{P}_t and an estimated initial template mesh \mathcal{X}_0 , we apply grid-based preconditioning to diffuse the under-constrained gradient information during optimization of the transformations $\mathbf{T}_t^{l,c}$ within the grid.

stacked transformation parameters for all grid cells at resolution level l . A standard gradient descent update is given by:

$$\mathbf{T}^l \leftarrow \mathbf{T}^l - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{T}^l}, \quad (2)$$

where η is the learning rate and \mathcal{L} the loss function. Instead, we apply spatially-aware preconditioning [NJJ21] that couples updates between neighboring grid cells:

$$\mathbf{T}^l \leftarrow \mathbf{T}^l - \eta (\mathbf{I} + \lambda \mathbf{L}^l)^{-2} \frac{\partial \mathcal{L}}{\partial \mathbf{T}^l}, \quad (3)$$

where \mathbf{L}^l is the Laplacian matrix encoding adjacency between neighboring grid cells, and $\lambda > 0$ controls the strength of spatial smoothing. This corresponds to a heat diffusion process on the grid where, in this case, gradients between neighboring cells are continuously smoothed over time and λ determines the time scale of diffusion and thus the degree of smoothing.

This preconditioning approach offers several advantages over traditional energy-based regularization, as 1) the *elimination of the accuracy tradeoff* inherent in explicit regularization formulations, allowing the transformations to fit the data while still enforcing smoothness. By embedding smoothness in the optimization trajectory (rather than the final objective) serves as 2) *adaptive regularization* and preserves the ability to represent non-smooth deformations, such as sharp boundaries, whenever the data demand them. Furthermore, in under-determined problems, where multiple solutions exist that fit the data equally well, the optimization naturally leads toward 3) *smooth solutions*, thereby resolving ambiguities in a principled manner. Additionally, it improves 4) *numerical stability* by dampening high-frequency oscillations, leading to more robust convergence compared to standard gradient descent. Finally,

because the grid topology is fixed, the Laplacian matrix \mathbf{L}^l can be precomputed once, and its sparse structure exploited by standard linear solvers, leading to 5) an *efficient computation*.

Note that we apply preconditioning not only to our grids, but also to the optimization of the mesh vertices \mathcal{X}_0 (see Sec. 3.4), following the mesh preconditioning strategy by Nicolet et al. [NJJ21].

3.3. Transformation Parameterization

As mentioned in Sec. 3.1, each grid cell encodes a 6D transformation vector:

$$\mathbf{T}_t^{l,c} = [\mathbf{z}^T, \mathbf{t}^T]^T \quad (4)$$

where $\mathbf{z} = (z_0, z_1, z_2)^T \in \mathbb{R}^3$ parameterizes the rotation component and $\mathbf{t} = (t_x, t_y, t_z)^T \in \mathbb{R}^3$ specifies the translation component. To avoid issues such as gimbal lock and to ensure robust optimization, we adopt the Cayley parameterization [ZJA21] to express rotations. Specifically, the rotation matrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ is defined as:

$$\mathbf{R}(\mathbf{z}) = (\mathbf{I} + \mathbf{Z})(\mathbf{I} - \mathbf{Z})^{-1} \quad (5)$$

where \mathbf{Z} is the skew-symmetric matrix constructed from \mathbf{z} :

$$\mathbf{Z} = \begin{pmatrix} 0 & -z_2 & z_1 \\ z_2 & 0 & -z_0 \\ -z_1 & z_0 & 0 \end{pmatrix} \quad (6)$$

The full transformation at a point is represented by a 4×4 homogeneous matrix, composed of the interpolated rotation $\mathbf{R}(\mathbf{z})$ and translation \mathbf{t} :

$$\mathbf{M}(\hat{\mathbf{T}}_t(\mathbf{p})) = \begin{bmatrix} \mathbf{R}(\mathbf{z}) & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (7)$$

To estimate the deformed surface $\hat{\mathcal{X}}_t$ at time t , we apply the estimated transformations recursively from $t = 0$ up to the current step to the initial mesh \mathcal{X}_0 :

$$\hat{\mathcal{X}}_t = \left\{ \hat{\mathbf{x}}_t \in \mathbb{R}^3 \mid \hat{\mathbf{x}}_t = \prod_{\tau=1}^t \mathbf{M}(\hat{\mathbf{T}}_\tau(\mathbf{x}_{\tau-1})) \cdot \mathbf{x}_0 \right\} \quad (8)$$

For brevity, we omit the explicit conversion to and from homogeneous coordinates.

While the formulation above propagates transformations forward in time, that is from $\hat{\mathcal{X}}_{t-1}$ to $\hat{\mathcal{X}}_t$, the same mechanism can be applied in reverse. This allows more a flexible starting point by selecting a suitable keyframe for \mathcal{X}_0 at any time step $t > 0$ of the input point cloud sequence (see Sec. 3.5).

3.4. Optimization Objectives

Our method jointly optimizes the initial surface mesh \mathcal{X}_0 and the multi-resolution transformation grids $\{\mathbf{T}_t^{l,c}\}$ by minimizing:

$$\mathcal{L} = \mathcal{L}_{\text{mesh}} + \mathcal{L}_{\text{transform}} + w_{\text{isometry}} \cdot \mathcal{L}_{\text{isometry}}, \quad (9)$$

where $w_{\text{isometry}} = 250$ controls the contribution of the isometry loss $\mathcal{L}_{\text{isometry}}$. This value is chosen such that the isometry loss contributes only weakly, accounting for approximately 25% of the typical magnitude of the transformation loss $\mathcal{L}_{\text{transform}}$.

Surface Initialization Loss. To ensure accurate surface geometry at the keyframe, we align the initially estimated mesh \mathcal{X}_0 to the corresponding reference point cloud \mathcal{P}_0 by minimizing a robust variant of the Chamfer distance:

$$\mathcal{L}_{\text{mesh}} = \text{CD}_R(\mathcal{X}_0, \mathcal{P}_0), \quad (10)$$

where CD_R denotes the robust Chamfer distance [YRH*24], which is designed to reduce sensitivity to outliers:

$$\begin{aligned} \text{CD}_R(\mathcal{P}, \mathcal{Q}) = & \frac{1}{|\mathcal{P}|} \sum_{\mathbf{p} \in \mathcal{P}} w_R(\mathbf{p}, \mathbf{q}_p) \|\mathbf{p} - \mathbf{q}_p\|^2 \\ & + \frac{1}{|\mathcal{Q}|} \sum_{\mathbf{q} \in \mathcal{Q}} w_R(\mathbf{p}_q, \mathbf{q}) \|\mathbf{p}_q - \mathbf{q}\|^2 \end{aligned} \quad (11)$$

where

$$\mathbf{q}_p = \arg \min_{\mathbf{q} \in \mathcal{Q}} \|\mathbf{p} - \mathbf{q}\|, \quad \mathbf{p}_q = \arg \min_{\mathbf{p} \in \mathcal{P}} \|\mathbf{p} - \mathbf{q}\| \quad (12)$$

are the nearest neighbors of the points \mathbf{p} and \mathbf{q} in the opposite point sets \mathcal{Q} and \mathcal{P} , and

$$w_R(\mathbf{p}, \mathbf{q}) = \exp(-\alpha \cdot \|\mathbf{p} - \mathbf{q}\|^2) \quad (13)$$

is a robust weighting function $w_R(\mathbf{p}, \mathbf{q})$ defined as a Gaussian kernel with $\alpha = 5.56$. This formulation attenuates the influence of outliers by down-weighting correspondences with large residuals, thereby enhancing the robustness of the surface alignment.

Transformation Fitting Loss. The primary data fitting objective encourages correct alignment between the transformed surface and the target point clouds:

$$\mathcal{L}_{\text{transform}} = \frac{1}{T} \sum_{t=1}^T w_{\text{confidence}}(t) \cdot \text{CD}_R(\hat{\mathcal{X}}_t, \mathcal{P}_t) + \text{CD}_R(\hat{\mathcal{P}}_t, \mathcal{P}_t), \quad (14)$$

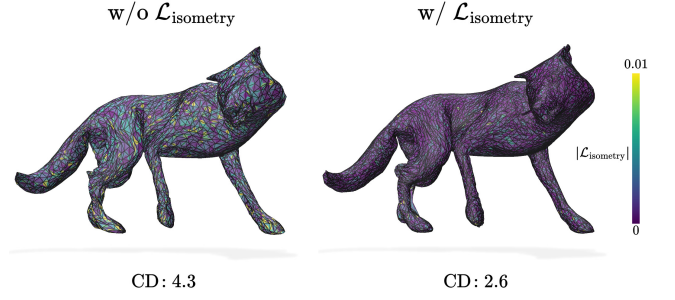


Figure 3: Effect of the complementary isometry loss $\mathcal{L}_{\text{isometry}}$ on the per-edge displacements. Without $\mathcal{L}_{\text{isometry}}$ (left), the surface is faithfully deformed but exhibits larger tangential deformations. When $\mathcal{L}_{\text{isometry}}$ is applied (right), the deformations become smoother and more temporally coherent, which also leads to improved Chamfer distances (CD) $[\times 10^{-5}]$.

where $\hat{\mathcal{P}}_t$ denotes the input points \mathcal{P}_{t-1} from time $t-1$ transformed forward by the transformation grid \mathcal{G}_t , and $w_{\text{confidence}}(t)$ is an adaptive confidence weight. This formulation leverages the grid structure to gather information from the input points at all timesteps, thereby guiding the mesh transformation process and enabling the estimation of robust, temporally coherent deformations.

Confidence-Based Error Control. Due to the sequential nature of our deformation model, inaccuracies in early transformations can propagate through time, forcing subsequent steps to compensate for accumulated errors. This often leads to increasingly large and unstable transformations, ultimately degrading reconstruction quality. To mitigate this, we introduce an adaptive confidence weighting scheme:

$$w_{\text{confidence}}(t) = \prod_{\tau=1}^t \left(\frac{1}{1 + \max(0, \text{CD}_R(\hat{\mathcal{X}}_\tau, \mathcal{P}_\tau) - \text{cd}_{\text{max}})} \right)^\delta, \quad (15)$$

which down-weights contributions from later frames if accumulated errors in preceding steps remain high. Here, the exponent δ is a schedule-dependent catch-up factor defined as:

$$\delta = 1 - \sqrt{e/e_{\text{max}}}, \quad (16)$$

where e denotes the current optimization epoch and e_{max} the total number of epochs. This ensures that $w_{\text{confidence}}(t)$ gradually increases toward 1 over time, allowing later frames to fully contribute once earlier deformations become sufficiently accurate, even for small residual errors over very long sequences. To normalize the confidence relative to the achievable alignment, we estimate a soft upper bound on reconstruction accuracy as:

$$\text{cd}_{\text{max}} = \max_{t \in [1:T]} \text{CD}_R(\text{sg}(\hat{\mathcal{P}}_t), \mathcal{P}_t), \quad (17)$$

where $\text{sg}(\cdot)$ denotes the stop-gradient operator, which prevents gradients from propagating through this term during backpropagation. This prevents feedback loops that could otherwise interfere with the learning of the deformation parameters $\mathbf{T}_t^{l,c}$.

Isometric Regularization. While the surface initialization and transformation fitting losses yield smooth and stable deformations, they do not enforce constraints along the mesh surface itself, including temporal coherence in these directions. However, in many real-world scenarios, surface motion is approximately isometric and thus intrinsic geometric properties such as edge lengths are typically preserved. To promote such physically plausible behavior, we introduce an isometry loss that penalizes temporal variations in edge length (see Fig. 3):

$$\mathcal{L}_{\text{isometry}} = \frac{1}{T|\mathcal{E}|} \sum_{t=1}^T \sum_{(i,j) \in \mathcal{E}} \left| \|\hat{\mathbf{x}}_{i,t} - \hat{\mathbf{x}}_{j,t}\| - \|\hat{\mathbf{x}}_{i,t-1} - \hat{\mathbf{x}}_{j,t-1}\| \right|, \quad (18)$$

where \mathcal{E} denotes the set of edges in the reference mesh topology.

3.5. Keyframe Selection

To initialize the optimization with a suitable initial surface, we follow a similar strategy as DynoSurf [YRH*24] to select a keyframe as the starting point. Rather than choosing the frame with the lowest aggregated Chamfer distance to all others, we prioritize selecting frames that exhibit a well-defined and representative surface topology. To this end, we measure the spatial extent of each frame and select the one with maximal coverage near the temporal midpoint:

$$t_{\text{key}} = \arg \max_{t \in [0:T]} w_{\text{key}} \left(t - \frac{T}{2} \right) |\mathcal{G}(\mathcal{P}_t)|, \quad (19)$$

where $|\mathcal{G}(\mathcal{P}_t)|$ denotes the number of occupied voxels in a fixed-resolution grid \mathcal{G} of size 128^3 . To favor a keyframe near the temporal center, we apply a Gaussian weighting function:

$$w_{\text{key}}(t) = \exp(-\gamma t^2), \quad \gamma = 0.001, \quad (20)$$

which discourages frames at the sequence boundaries, where it would be necessary to estimate larger overall transformations to other frames. Once t_{key} is determined, we reconstruct the initial mesh \mathcal{X}_0 via screened Poisson surface reconstruction [KH13] applied to the corresponding point cloud $\mathcal{P}_{t_{\text{key}}}$.

3.6. Implementation Details

All input point clouds \mathcal{P}_t are normalized to the spatial domain $[-1, 1]^3$ to align with the expected range of our multi-resolution voxel grid representation. To reduce computational overhead, we prune the transformation grid by retaining only cells that are either directly occupied by input points or fall within a three-cell neighborhood of occupied regions. This sparsification strategy reduces the number of active parameters by over 50%, substantially lowering memory consumption and accelerating optimization. As a result, our full pipeline typically requires less than 2 GB of GPU memory and completes processing a sequence in approximately 7 minutes on an NVIDIA RTX 4090. For optimization, we use the Adam optimizer with default hyperparameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$), combined with a hierarchical learning rate schedule across grid resolutions. The coarsest grid level ($l = 1$) is optimized using a base learning rate η of 5×10^{-3} , which is increased by 10% for each subsequent (finer) level. Similarly, the preconditioning smoothness weight λ is initialized at 0.25 for the coarsest level and increased by 50% per level to enforce stronger

Table 1: Quantitative comparison of reconstruction performance across common animation datasets. Our method consistently achieves the lowest Chamfer distance and Correspondence Error as well as the highest Normal Consistency and F-scores, demonstrating superior accuracy and temporal coherence compared to baseline methods.

		corresp.	CD [$\times 10^{-5}$] ↓	NC ↑	F-0.5% ↑	F-1% ↑	Corr. ↓
DFAUST	CaDeX	✗	3.68	0.941	0.730	0.921	0.539
	DynoSurf	✗	2.13	0.953	0.980	0.992	0.356
	M2V	✓	1.61	0.960	0.877	0.979	0.358
	Ours	✗	0.52	0.957	0.988	0.997	0.355
DT4D	CaDeX	✗	56.51	0.870	0.386	0.652	0.442
	DynoSurf	✗	15.18	0.919	0.773	0.922	0.419
	M2V	✓	7.61	0.944	0.792	0.938	0.425
	Ours	✗	1.53	0.960	0.961	0.994	0.422
AMA	DynoSurf	✗	1.01	0.918	0.921	0.992	0.347
	Ours	✗	0.47	0.939	0.985	0.999	0.348

spatial coherence at higher resolutions. The optimization of the mesh vertices of \mathcal{X}_0 is also preconditioned, using a fixed learning rate of 1×10^{-4} and a smoothness weight of $\lambda = 16$.

4. Evaluation

We evaluated our method on three animation datasets encompassing both human and animal motion. Specifically, AMA [VBMP08] and DFAUST [BRPMB17] comprise diverse human motion sequences, while DT4D [LTT*21] provides animal motions, offering broader coverage beyond human-centric benchmarks. For comparisons to learning-based methods, we adopted the dataset splits of DynoSurf [YRH*24], yielding 33 test sequences for AMA, 109 for DFAUST, and 89 for DT4D. We used the official checkpoints and code released by each method, without additional fine-tuning, and restricted the performance evaluations to the datasets each method was originally trained on for a fair comparison. For DynoSurf, we report both the published results for general benchmarking and results from additional experiments and ablations using the official implementation.

4.1. Comparison to State-of-the-Art

We compare our approach against three state-of-the-art methods for 4D surface reconstruction from point cloud sequences. The learning-based baselines CaDeX [LD22] and Motion2VecSets (M2V) [CLZ*24] leverage pre-trained models and incorporate category-specific priors, explicitly distinguishing between human and animal motion patterns. In contrast, our method and DynoSurf [YRH*24] are category-agnostic and, thus, do not require pre-training or semantic priors. Notably, M2V additionally assumes access to dense temporal point correspondences, introducing stronger requirements on input data and thereby relying on significantly more prior information. Reconstruction quality is evaluated using ℓ_2 -Chamfer Distance (CD), which quantifies geometric accuracy, and Normal Consistency (NC), which measures the alignment of surface normals across reconstructions. To assess temporal coherence, we report Correspondence Error (Corr.) and compute F-scores at 0.5% and 1% thresholds to quantify geometric overlap. All meshes are normalized to a unit bounding box within $[0, 1]^3$.

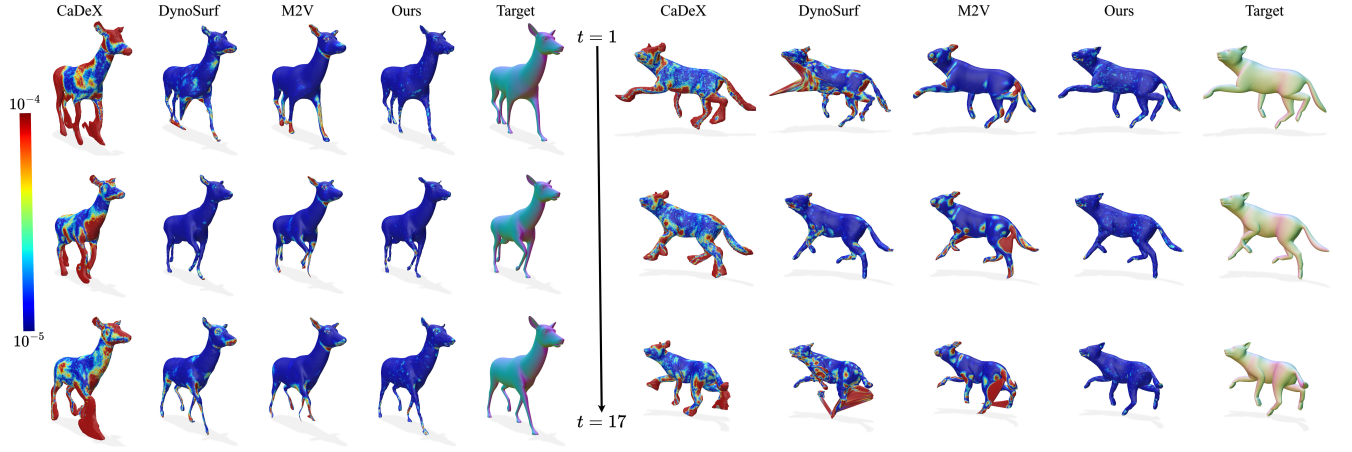


Figure 4: Comparison of visual results for CaDeX, DynoSurf, M2V, and Our method on two motion sequences of the DT4D dataset. Color maps indicate per-vertex ℓ_2 -Chamfer distance. Our method achieves the lowest error and best visual fidelity across all frames.

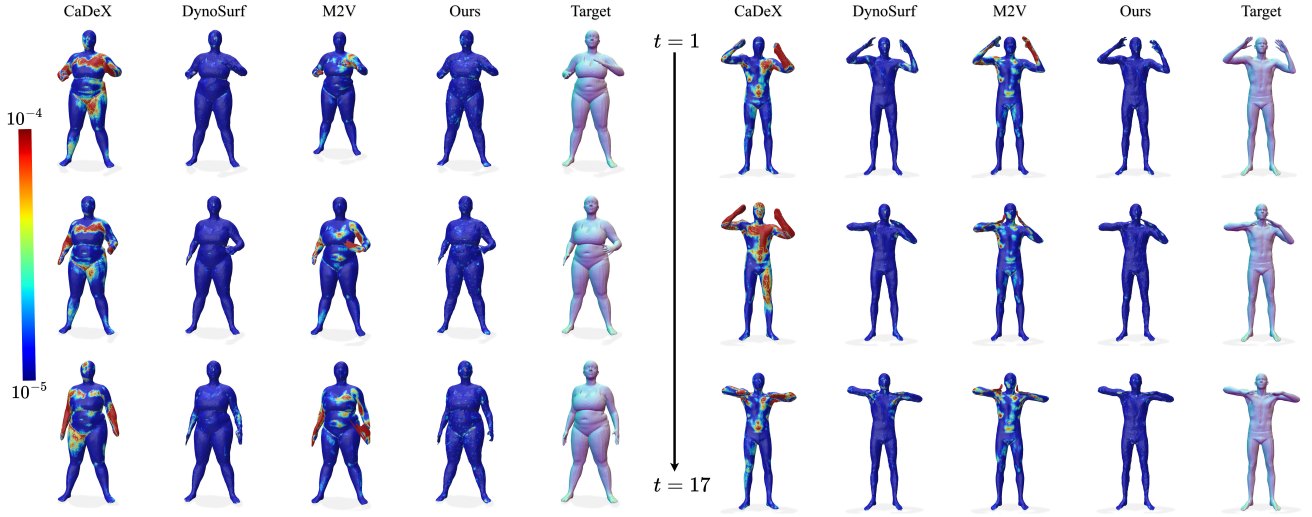


Figure 5: Comparison of visual results for CaDeX, DynoSurf, M2V, and Our method on two motion sequences of the DFAUST dataset. Color maps indicate per-vertex ℓ_2 -Chamfer distance. Our method achieves the lowest error and best visual fidelity across all frames.

prior to evaluation. For metric computation, 100000 points are uniformly sampled from both the predicted and ground-truth surfaces at each timestep. Unless otherwise stated, all experiments are performed with 5000 input target points per timestep over a motion sequence consisting of 17 timesteps. As summarized in Table 1, our method consistently outperforms baselines across all metrics and object categories, despite the absence of category-specific supervision. Qualitative results, visualized in Figs. 4 and 5, further highlight the effectiveness of our method, showing Chamfer distance maps on two DT4D animal sequences and two DFAUST human sequences, respectively.

Sequence Length Analysis. The accuracy of transformation estimation can vary significantly with the motion sequence length. To analyze this dependency, we evaluated the performance of our

Table 2: Performance across varying sequence lengths T on the AMA dataset. Our method remains robust as the length increases, while DynoSurf exhibits significant performance degradation.

T		CD [$\times 10^{-5}$] ↓	NC ↑	F-0.5% ↑	F-1% ↑
10	DynoSurf	0.80	0.919	0.943	0.996
	Ours	0.55	0.928	0.980	0.999
20	DynoSurf	2.22	0.886	0.809	0.961
	Ours	0.54	0.928	0.979	0.999
40	DynoSurf	4.64	0.857	0.686	0.902
	Ours	0.66	0.923	0.971	0.999
60	DynoSurf	14.64	0.778	0.474	0.743
	Ours	0.72	0.919	0.960	0.997
80	DynoSurf	16.32	0.759	0.431	0.700
	Ours	1.35	0.906	0.931	0.990

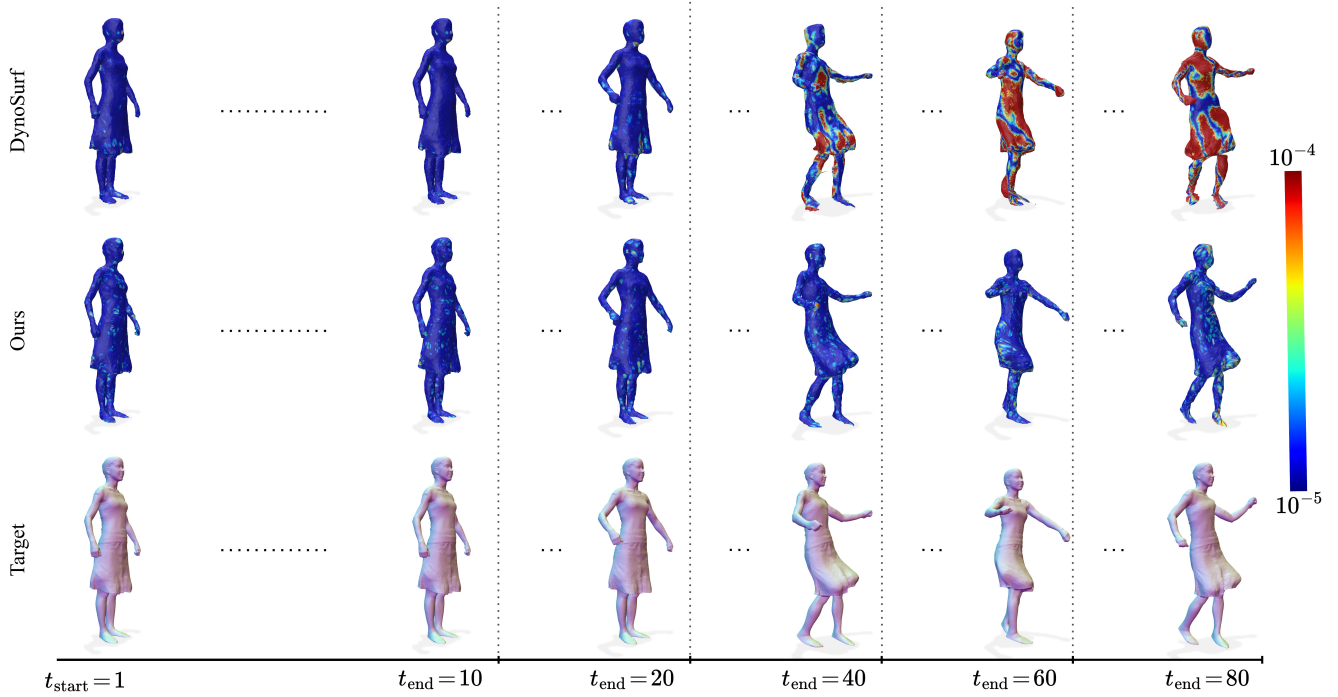


Figure 6: Comparison of CaDeX, DynoSurf, and Our method on the AMA dataset across motion sequences of increasing length, all beginning from the same initial pose. The final frame of each sequence is shown, with reconstruction error visualized as ℓ_2 -Chamfer distance, clamped to the range $[10^{-5}, 10^{-4}]$. Our method consistently maintains lower error over time, demonstrating robustness to sequence length.

method across sequences of varying lengths and compared it to alternative approaches that do not employ correlation-based mechanisms. This study is conducted on the AMA [VBMP08] dataset, which features complex, long-range human motions well-suited for assessing temporal robustness. For the analysis, the sequences are partitioned into fixed lengths of 10, 20, 40, 60, and 80 time frames. As shown in Fig. 6 and Table 2, we visualize the reconstructions at the final timestep of each segment, illustrating our method’s capability to capture extended temporal deformations with high accuracy. For longer sequences, we proportionally increase the number of optimization steps up to 10000 to account for our method’s frame-by-frame optimization strategy described in Sec. 3.3.

4.2. Ablation Study

To assess the contribution of each component in our pipeline, we conducted a series of ablation studies on the test split of the AMA [VBMP08] dataset. In particular, we examined the method’s sensitivity to input noise, the effect of varying the number of levels in the multi-resolution grid, and the impact of components like smoothness preconditioning and the isometry loss. These studies offer insights into how specific design choices influence reconstruction accuracy, robustness to noise, and optimization stability.

Point Cloud Resolution. For methods that do not leverage pre-trained knowledge of the transformed surfaces, the resolution of the target point clouds plays a critical role in reconstruction perfor-

Table 3: Performance at varying point cloud resolutions. Our method consistently improves with higher input densities, while DynoSurf shows limited scalability.

$ \mathcal{P}_t $		CD $[\times 10^{-5}] \downarrow$	NC \uparrow	F-0.5% \uparrow	F-1% \uparrow
2500	DynoSurf	1.56	0.896	0.862	0.981
	Ours	0.79	0.922	0.948	0.997
5000	DynoSurf	1.01	0.918	0.921	0.992
	Ours	0.47	0.939	0.985	0.999
10000	DynoSurf	1.28	0.906	0.897	0.986
	Ours	0.40	0.950	0.993	0.999
20000	DynoSurf	1.45	0.902	0.887	0.982
	Ours	0.37	0.959	0.996	0.999

mance. Increasing the number of target points can also help expose the performance ceiling of such approaches. To investigate this, we evaluated each method using target point clouds ranging from 2500 to 20000 points. As shown in Table 3, our method scales effectively with resolution, while alternative methods exhibit early saturation in performance at lower resolutions. In addition, surface normals and object-specific normal consistency for the reconstructed geometries are visualized in Fig. 7.

Key Components. We conducted an ablation study to evaluate the contributions of three key components of our grid optimization: the multi-resolution grid structure, smoothness preconditioning of the

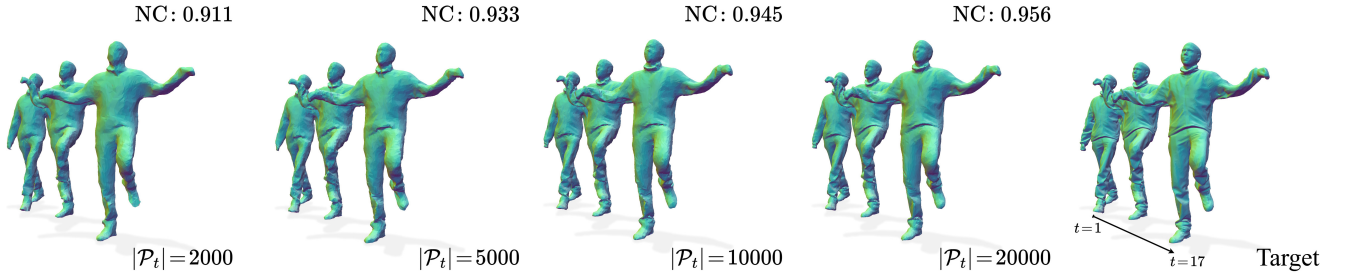


Figure 7: Effect of point cloud resolution on reconstruction quality. Our method remains robust across varying numbers of input points, maintaining high accuracy even at lower resolutions.

Table 4: Ablation on key components of our method. When the multi-resolution voxel grid is disabled, only the finest resolution level is used. Additionally, in the absence of preconditioning, the learning rate of the grid is reduced to 10%. Results show that the multi-resolution grid alone yields the largest gain in Chamfer distance, while preconditioning improves F-scores and normal consistency. The full model achieves the best overall performance.

Smooth. Prec.	Multi-Res.	Isometry	CD [$\times 10^{-5}$] ↓	NC ↑	F-0.5% ↑	F-1% ↑
-	-	-	9×10^5	0.796	0.720	0.791
-	-	✓	3×10^2	0.901	0.929	0.967
-	✓	-	0.69	0.910	0.960	0.997
-	✓	✓	4.39	0.913	0.958	0.991
✓	-	-	2.54	0.901	0.950	0.991
✓	-	✓	4.37	0.913	0.958	0.991
✓	✓	-	0.67	0.933	0.978	0.998
✓	✓	✓	0.47	0.939	0.985	0.999

grid cells, and the isometry loss. To ensure a fair comparison, we reduced the grid learning rate to 10% of its original value when smoothness preconditioning is disabled, as this component is essential for stable optimization at higher step sizes. As shown in Table 4, only enabling the multi-resolution grid already leads to the most substantial performance improvement as it allows to model deformations across multiple spatial scales. Adding smoothness preconditioning further enhances reconstruction quality by encouraging stable, coherent transformation updates during optimization. Its impact is most apparent when combined with the multi-resolution grid, particularly in the surface metrics, normal consistency and F-scores, by promoting smooth motion of neighboring regions and thereby improving surface reconstruction quality. While the isometry loss contributes less in terms of quantitative metrics, it plays an important role in constraining local surface distortions and preserving temporal consistency, particularly in near-static or under-constrained regions.

Noise. We evaluated the robustness of our method to input noise by adding Gaussian noise to the point clouds. The noise magnitude is chosen based on the bounding box diagonal of the input point cloud, with levels set to 0.25%, 0.5%, 1%, and 2%. As shown in Table 5, our method maintains high reconstruction quality under moderate noise, despite not employing any explicit denoising strat-

Table 5: Reconstruction performance under varying levels of Gaussian noise added to the input point clouds. The noise magnitude is expressed as a percentage of the bounding box diagonal. Results demonstrate our method’s robustness, with stable reconstruction quality observed across moderate noise levels.

Noise	CD [$\times 10^{-5}$] ↓	NC ↑	F-0.5% ↑	F-1% ↑
0%	0.47	0.939	0.985	0.999
0.25%	0.76	0.912	0.958	0.998
0.5%	1.25	0.865	0.870	0.995
1%	3.83	0.703	0.581	0.902
2%	20.90	0.529	0.254	0.508

egy. While performance degrades gradually with increasing noise, the results remain stable up to 1%, demonstrating the strong resilience of our method to imperfect input. Significant degradation is observed only at the extreme noise level of 2% which, however, represents an unlikely real-world use case.

Grid Size. We also investigated the impact of the grid resolution by varying the number of levels in our hierarchical voxel structure. Specifically, we compare configurations with 1, 2, 4, 6, 8, and 12 levels against our default setup with 10 levels. As illustrated in Fig. 8, even low-resolution configurations with significantly fewer grid cells are capable of providing high-quality reconstructions. Increasing the number of levels consistently improves the reconstruction accuracy by capturing finer-scale local deformations. However, gains beyond 8 levels become increasingly marginal, while the default configuration of 10 levels strikes a good balance between accuracy and computational cost. Crucially, the added flexibility of higher resolutions (12 levels) does not degrade performance, demonstrating the stability and scalability of our approach.

Initialization. We evaluated how keyframe selection t_{key} and surface initialization affect reconstruction quality. Specifically, we compared 1) choosing the first frame, 2) the temporal middle frame, and 3) our coverage-weighted keyframe (see Sec. 3.5), as well as 1) screened Poisson surface reconstruction [KH13], 2) the deformable tetrahedron of DynoSurf [YRH*24], and 3) a pretrained diffusion initializer [CLZ*24]. As shown in Table 6, Poisson reconstruction consistently achieved robust results on the DT4D dataset and is

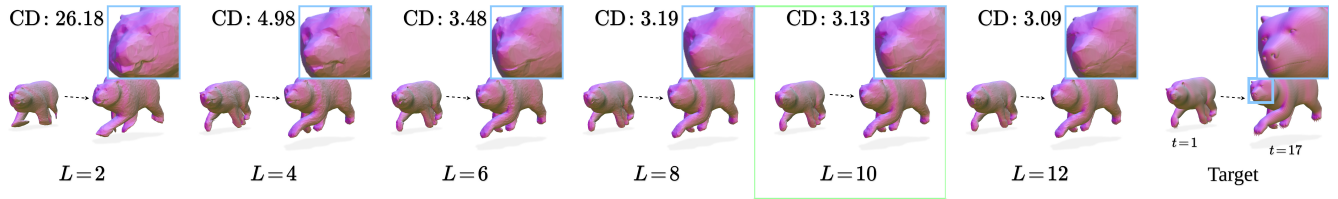


Figure 8: Effect of the number of grid levels on reconstruction quality. We report the Chamfer distance (CD) $[10^{-5}]$ to highlight reconstruction accuracy. Our default configuration with 10 levels is marked in green, while zoom-ins on key regions are shown in blue. Fewer levels already yield reasonable results, but additional levels enhance fine-scale detail without compromising stability.

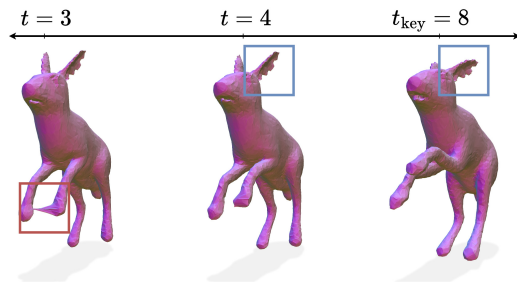


Figure 9: Failure cases of our method. Blue regions highlight artifacts caused by sparse input sampling at the initial timestep, leading to errors in the initial surface reconstruction. Red regions indicate correspondence errors resulting from insufficient alignment of the input points during grid optimization.

insensitive to the keyframe choice. The diffusion initializer may achieve higher geometric accuracy when seeded with a favorable keyframe, but it highly relies on good keyframes t_{key} and quickly degrades otherwise. In general, both the middle and our selection scheme yield small, yet reliable improvements over picking the first frame. Considering the AMA dataset, where diffusion would require retraining and is therefore excluded, Poisson reconstruction also outperforms the tetrahedron baseline. Similarly, choosing the middle frame or our coverage-based keyframe consistently improves the metrics compared to the first frame. Here, our scheme typically matches the middle frame very closely in practice, making it a reliable proxy for the unknown optimal frame. Especially for the tetrahedron initialization, this leads to substantial gains and narrows down the gap to Poisson reconstruction. Overall, Poisson reconstruction offered the best trade-off between accuracy and stability, especially when paired with our keyframe selection.

4.3. Limitations

While our method demonstrates strong robustness across a range of scenarios, certain failure cases remain, as illustrated in Fig. 9. One limitation arises when the point sampling at the keyframe is too sparse. In this case, the initial surface reconstruction based on Laplacian regularization may exhibit artifacts that persist throughout the sequence. This occurs because the surface mesh is optimized solely to match the sparse keyframe point cloud, which

Table 6: We compare the behavior of our method when selecting the first or middle frame as the keyframe, against our proposed keyframe selection strategy. In addition, we evaluate different point cloud-to-surface initialization methods, comparing our screened Poisson reconstruction to the deformable tetrahedron approach [YRH*24] and the pretrained diffusion model [CLZ*24].

	t_{key}	Method	CD $[\times 10^{-5}] \downarrow$	NC \uparrow	F-0.5% \uparrow	F-1% \uparrow
DT4D	First	Diffusion	8.42	0.956	0.942	0.984
		Tetrahedron	5.07	0.912	0.879	0.940
		Poisson	2.50	0.959	0.951	0.992
	Middle	Diffusion	1.87	0.962	0.956	0.992
		Tetrahedron	6.70	0.925	0.902	0.948
		Poisson	2.32	0.962	0.961	0.994
	Ours	Diffusion	3.53	0.960	0.954	0.989
		Tetrahedron	4.45	0.925	0.905	0.954
		Poisson	2.35	0.962	0.961	0.994
AMA	First	Tetrahedron	0.60	0.930	0.973	0.998
		Poisson	0.59	0.931	0.975	0.999
	Middle	Tetrahedron	0.60	0.933	0.977	0.998
		Poisson	0.52	0.935	0.981	0.999
	Ours	Tetrahedron	0.54	0.934	0.978	0.999
		Poisson	0.53	0.935	0.981	0.999

may lack sufficient detail to constrain the geometry accurately. The transformation grid cannot resolve these artifacts either, as the corresponding erroneous regions are represented by too few points in subsequent timesteps to trigger corrective deformations. Another limitation involves occasional errors in transformation estimation, particularly when the grid is not sufficiently optimized with respect to the input. This can lead to inaccurate local correspondences, which propagate over time and degrade alignment quality. These issues are partially influenced by the confidence scaling term, which controls the influence of previous steps in the optimization.

5. Conclusion

We introduced Preconditioned Deformation Grids, a correspondence-free and training-free technique for estimating coherent deformation fields directly from unstructured point cloud sequences. Our method addressed the inherently under-constrained nature of this problem by employing Sobolev preconditioning, which spatially diffuses gradient information to achieve a spatially adaptive smoothness. We further guided the optimization using multi-resolution voxel grids to represent the deformation field, allowing coarser levels to maintain temporal coherence over long

sequences and finer levels to capture high-frequency surface details. Through extensive qualitative and quantitative experiments, we demonstrated that our method achieves superior reconstruction results over existing methods using only a simple Chamfer loss and a weak isometry loss, providing a robust and flexible solution for arbitrary object motion without relying on restrictive priors or extensive training data.

Future Work. Our framework offers a solid foundation for several natural extensions. Beyond point clouds, adapting it to richer representations such as 3D Gaussian Splatting or implicit neural primitives could broaden its applicability to 4D capturing scenarios. Moreover, modifying the grid structure toward physically motivated formulations may enable the estimation of complex dynamics including fluid motion, thereby bringing reconstruction and simulation closer together. Finally, exploring adaptive or learned preconditioning strategies could further improve robustness by adjusting smoothness to local data characteristics.

Acknowledgements

This research has been funded by the Federal Ministry of Education and Research under grant no. 01IS22094A WEST-AI, by the Federal Ministry of Education and Research of Germany as well as the state of North-Rhine Westphalia as part of the Lamarr-Institute for Machine Learning and Artificial Intelligence, by the Ministry of Culture and Science North Rhine-Westphalia under grant number PB22-063A (InVirtuo 4.0: Experimental Research in Virtual Environments) and by the state of North Rhine-Westphalia as part of the Excellency Start-up Center.NRW (U-BO-GROW) under grant number 03ESCNW18B. Open Access funding enabled and organized by Projekt DEAL.

References

- [AMB*19] ALLDIECK T., MAGNOR M., BHATNAGAR B. L., THEOBALT C., PONS-MOLL G.: Learning to Reconstruct People in Clothing from a Single RGB Camera. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019). 2
- [ASK*05] ANGUELOV D., SRINIVASAN P., KOLLER D., THRUN S., RODGERS J., DAVIS J.: SCAPE: Shape Completion and Animation of People. *ACM Transactions on Graphics (TOG)* 24, 3 (2005). 2
- [BKL*16] BOGO F., KANAZAWA A., LASSNER C., GEHLER P., ROMERO J., BLACK M. J.: Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In *European Conference on Computer Vision (ECCV)* (2016). 2
- [BPZ*20] BOZIC A., PALAFOX P., ZOLLHÖFER M., DAI A., THIES J., NIESSNER M.: Neural Non-Rigid Tracking. *Advanced Neural Information Processing Systems (NeurIPS)* 33 (2020). 2
- [BPZ*21] BOZIC A., PALAFOX P., ZOLLHOFER M., THIES J., DAI A., NIESSNER M.: Neural Deformation Graphs for Globally-consistent Non-rigid Reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021). 2
- [BRPMB17] BOGO F., ROMERO J., PONS-MOLL G., BLACK M. J.: Dynamic FAUST: Registering Human Bodies in Motion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2017). 6
- [BTTPM19] BHATNAGAR B. L., TIWARI G., THEOBALT C., PONS-MOLL G.: Multi-Garment Net: Learning to Dress 3D People from Images. In *IEEE International Conference on Computer Vision (ICCV)* (2019). 2
- [CBSS17] CLAICI S., BESSMELTSEV M., SCHAEFER S., SOLOMON J.: Isometry-Aware Preconditioning for Mesh Parameterization. In *Computer Graphics Forum (CGF)* (2017), vol. 36. 3
- [Che25] CHEN J.: Graph Neural Preconditioners for Iterative Solutions of Sparse Linear Systems. In *International Conference on Learning Representations (ICLR)* (2025). 3
- [CLZ*24] CAO W., LUO C., ZHANG B., NIESSNER M., TANG J.: Motion2VecSets: 4D Latent Vector Set Diffusion for Non-rigid Shape Reconstruction and Tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024). 3, 6, 9, 10
- [CRBD18] CHEN R. T., RUBANOVA Y., BETTENCOURT J., DUVE-NAUD D. K.: Neural Ordinary Differential Equations. *Advanced Neural Information Processing Systems (NeurIPS)* 31 (2018). 2
- [CYB*24] CHANG W., YANG X., BELHE Y., RAMAMOORTHY R., LI T.-M.: Spatiotemporal Bilateral Gradient Filtering for Inverse Rendering. In *SIGGRAPH Asia Conference Papers* (2024). 3
- [EPT*07] ECKSTEIN I., PONS J.-P., TONG Y., KUO C.-C., DESBRUN M.: Generalized Surface Flows for Mesh Processing. In *Eurographics Symposium on Geometry Processing (SGP)* (2007). 3
- [HGA*23] HUANG J., GOJCIC Z., ATZMON M., LITANY O., FIDLER S., WILLIAMS F.: Neural Kernel Surface Reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023). 1
- [HJL*20] HUANG J., JIANG C. M., LENG B., WANG B., GUIBAS L.: MeshODE: A Robust and Scalable Framework for Mesh Deformation. *arXiv preprint arXiv:2005.11617* (2020). 3
- [HMGCO20] HANOCA R., METZER G., GIRYES R., COHEN-OR D.: Point2Mesh: A Self-Prior for Deformable Meshes. *ACM Transactions on Graphics (TOG)* 39, 4 (2020). 1
- [HÖS24] HÄUSNER P., ÖKTEM O., SJÖLUND J.: Neural incomplete factorization: learning preconditioners for the conjugate gradient method. *Transactions on Machine Learning Research* (2024). 3
- [JKYL25] JUNG Y., KIM H., YOON H., LEE S.: Preconditioned Single-step Transforms for Non-rigid ICP. In *Computer Graphics Forum (CGF)* (2025). 3
- [JZW*21] JIANG B., ZHANG Y., WEI X., XUE X., FU Y.: Learning Compositional Representation for 4D Captures with Neural ODE. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021). 3
- [KFS13] KRISHNAN D., FATTAL R., SZELISKI R.: Efficient Preconditioning of Laplacian Matrices for Computer Graphics. *ACM Transactions on Graphics (TOG)* 32, 4 (2013). 3
- [KGL16] KOVALSKY S. Z., GALUN M., LIPMAN Y.: Accelerated Quadratic Proxy for Geometric Optimization. *ACM Transactions on Graphics (TOG)* 35, 4 (2016). 3
- [KH13] KAZHDAN M., HOPPE H.: Screened Poisson Surface Reconstruction. *ACM Transactions on Graphics (TOG)* 32, 3 (2013). 6, 9
- [KKLD23] KERBL B., KOPANAS G., LEIMKÜHLER T., DRETTAKIS G.: 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (TOG)* 42, 4 (2023). 2
- [LBB*17] LI T., BOLKART T., BLACK M. J., LI H., ROMERO J.: Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics (TOG)* 36, 6 (2017). 2
- [LCDM23] LI Y., CHEN P. Y., DU T., MATUSIK W.: Learning Preconditioners for Conjugate Gradient PDE Solvers. In *International Conference on Machine Learning (ICML)* (2023). 3
- [LD22] LEI J., DANILIDIS K.: CaDeX: Learning Canonical Deformation Coordinate Space for Dynamic Surface Representation via Neural Homeomorphism. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022). 2, 3, 6
- [LH22] LI Y., HARADA T.: Non-rigid Point Cloud Registration with Neural Deformation Pyramid. *Advanced Neural Information Processing Systems (NeurIPS)* 35 (2022). 2

- [LHR*21] LIU L., HABERMANN M., RUDNEV V., SARKAR K., GU J., THEOBALT C.: Neural Actor: Neural Free-view Synthesis of Human Actors with Pose Control. *ACM Transactions on Graphics (TOG)* 40, 6 (2021). 2
- [LMR*15] LOPER M., MAHMOOD N., ROMERO J., PONS-MOLL G., BLACK M. J.: SMPL: A Skinned Multi-Person Linear Model. *ACM Transactions on Graphics (TOG)* 34, 6 (2015). 2
- [LSNW21] LI Z., NIKLAUS S., SNAVELY N., WANG O.: Neural Scene Flow Fields for Space-Time View Synthesis of Dynamic Scenes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021). 3
- [LSS*25] LI R. B., SHAGHAGHI M., SUZUKI K., LIU X., MOPARTHI V., DU B., CURTIS W., RENSCHLER M., LEE K. M. B., ATANASOV N., NGUYEN T.: Dynaglam: Real-time gaussian-splatting slam for online rendering, tracking, motion predictions of moving objects in dynamic scenes, 2025. 2
- [LTT*21] LI Y., TAKEHARA H., TAKETOMI T., ZHENG B., NIESSNER M.: 4DComplete: Non-Rigid Motion Estimation Beyond the Observable Surface. In *IEEE International Conference on Computer Vision (ICCV)* (2021). 6
- [MBD25] MATSUKI H., BAE G., DAVISON A. J.: 4dtam: Non-rigid tracking and mapping via dynamic surface gaussians. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2025). 2
- [MJB13] MARTIN T., JOSHI P., BERGOU M., CARR N.: Efficient non-linear optimization via multi-scale gradient filtering. In *Computer Graphics Forum (CGF)* (2013), vol. 32. 3
- [Neu85] NEUBERGER J.: Steepest descent and differential equations. *Journal of the Mathematical Society of Japan* 37, 2 (1985). 3
- [NFS15] NEWCOMBE R. A., FOX D., SEITZ S. M.: DynamicFusion: Reconstruction and Tracking of Non-rigid Scenes in Real-Time. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2015). 2
- [NJJ21] NICOLET B., JACOBSON A., JAKOB W.: Large Steps in Inverse Rendering of Geometry. *ACM Transactions on Graphics (TOG)* 40, 6 (2021). 3, 4
- [NLW*25] NIZAMANI A., LAGA H., WANG G., BOUSSAID F., BEN-NAMOUN M., SRIVASTAVA A.: Dynamic neural surfaces for elastic 4d shape representation and analysis, 2025. 3
- [NMOG19] NIEMEYER M., MESCHERER L., OECHSLE M., GEIGER A.: Occupancy Flow: 4D Reconstruction by Learning Particle Dynamics. In *IEEE International Conference on Computer Vision (ICCV)* (2019). 2
- [PCG*19] PAVLAKOS G., CHOUTAS V., GHORBANI N., BOLKART T., OSMAN A. A., TZIONAS D., BLACK M. J.: Expressive Body Capture: 3D Hands, Face, and Body from a Single Image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019). 2
- [PCPMN21] PUMAROLA A., CORONA E., PONS-MOLL G., MORENO-NOGUER F.: D-NeRF: Neural Radiance Fields for Dynamic Scenes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021). 3
- [PJL*21] PENG S., JIANG C., LIAO Y., NIEMEYER M., POLLEFEYS M., GEIGER A.: SShape As Points: A Differentiable Poisson Solver. *Advanced Neural Information Processing Systems (NeurIPS)* 34 (2021). 1
- [PSB*21] PARK K., SINHA U., BARRON J. T., BOUAZIZ S., GOLDMAN D. B., SEITZ S. M., MARTIN-BRUALLA R.: Nerfies: Deformable Neural Radiance Fields. In *IEEE International Conference on Computer Vision (ICCV)* (2021). 3
- [Ren04] RENKA R. J.: Constructing fair curves and surfaces with a Sobolev gradient method. *Computer Aided Geometric Design* 21, 2 (2004). 3
- [RFM*24] RUDIKOV A., FANASKOV V., MURAVLEVA E., LAEVSKY Y. M., OSELEDETS I.: Neural operators meet conjugate gradients: The FCG-NO method for efficient PDE solving. In *International Conference on Machine Learning (ICML)* (2024). 3
- [RN95] RENKA R. J., NEUBERGER J.: Minimal Surfaces and Sobolev Gradients. *SIAM Journal on Scientific Computing* 16, 6 (1995). 3
- [SBFB19] SANYAL S., BOLKART T., FENG H., BLACK M. J.: Learning to Regress 3D Face Shape and Expression from an Image without 3D Supervision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019). 2
- [SBI18] SLAVCHEVA M., BAUST M., ILIC S.: SobolevFusion: 3D Reconstruction of Scenes Undergoing Free Non-rigid Motion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2018). 3
- [SSP07] SUMNER R. W., SCHMID J., PAULY M.: Embedded deformation for shape manipulation. *ACM Transactions on Graphics (TOG)* 26, 3 (2007). 2
- [TBTPM20] TIWARI G., BHATNAGAR B. L., TUNG T., PONS-MOLL G.: SIZER: A Dataset and Model for Parsing 3D Clothing and Learning Size Sensitive 3D Clothing. In *European Conference on Computer Vision (ECCV)* (2020). 2
- [TRI*24] TRIFONOV V., RUDIKOV A., ILIEV O., LAEVSKY Y. M., OSELEDETS I., MURAVLEVA E.: Learning from Linear Algebra: A Graph Neural Network Approach to Preconditioner Design for Conjugate Gradient Solvers. *arXiv preprint arXiv:2405.15557* (2024). 3
- [TXJZ21] TANG J., XU D., JIA K., ZHANG L.: Learning Parallel Dense Correspondence from Spatio-Temporal Descriptors for Efficient and Robust 4D Reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021). 3
- [VBMP08] VLASIC D., BARAN I., MATUSIK W., POPOVIC J.: Articulated Mesh Animation from Multi-view Silhouettes. *ACM Transactions on Graphics (TOG)* 27, 3 (2008). 6, 8
- [WCY*22] WANG L., CHEN Z., YU T., MA C., LI L., LIU Y.: FaceVerse: a Fine-grained and Detail-controllable 3D Face Morphable Model from a Hybrid Dataset. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022). 2
- [WLLY19] WU Z., LI K., LAI Y.-K., YANG J.: Global as-Conformal-as-Possible Non-Rigid Registration of Multi-view Scans. In *IEEE International Conference on Multimedia and Expo (ICME)* (2019), IEEE. 2
- [WSS*19] WILLIAMS F., SCHNEIDER T., SILVA C., ZORIN D., BRUNA J., PANOZZO D.: Deep Geometric Prior for Surface Reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019). 1
- [XBZ*20] XU H., BAZAVAN E. G., ZANFIR A., FREEMAN W. T., SUKTHANKAR R., SMINCISESCU C.: GHUM & GHUML: Generative 3D Human Shape and Articulated Pose Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020). 2
- [YGL*19] YANG J., GUO D., LI K., WU Z., LAI Y.-K.: Global 3D Non-Rigid Registration of Deformable Objects Using a Single RGB-D Camera. *IEEE Transactions on Image Processing* 28, 10 (2019). 2
- [YRH*24] YAO Y., REN S., HOU J., DENG Z., ZHANG J., WANG W.: DynoSurf: Neural Deformation-based Temporally Consistent Dynamic Surface Reconstruction. In *European Conference on Computer Vision (ECCV)* (2024). 3, 5, 6, 9, 10
- [YTB*21] YENAMANDRA T., TEWARI A., BERNARD F., SEIDEL H.-P., ELGHARIB M., CREMERS D., THEOBALT C.: i3DMM: Deep Implicit 3D Morphable Model of Human Heads. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021). 3
- [ZJA21] ZHANG J. E., JACOBSON A., ALEXA M.: Fast Updates for Least-Squares Rotational Alignment. *Computer Graphics Forum (CGF)* 40, 2 (2021). 4
- [ZZB*25] ZHENG J., ZHU Z., BIERI V., POLLEFEYS M., PENG S., ARMENI I.: Wildgs-slam: Monocular gaussian splatting slam in dynamic environments. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2025). 2