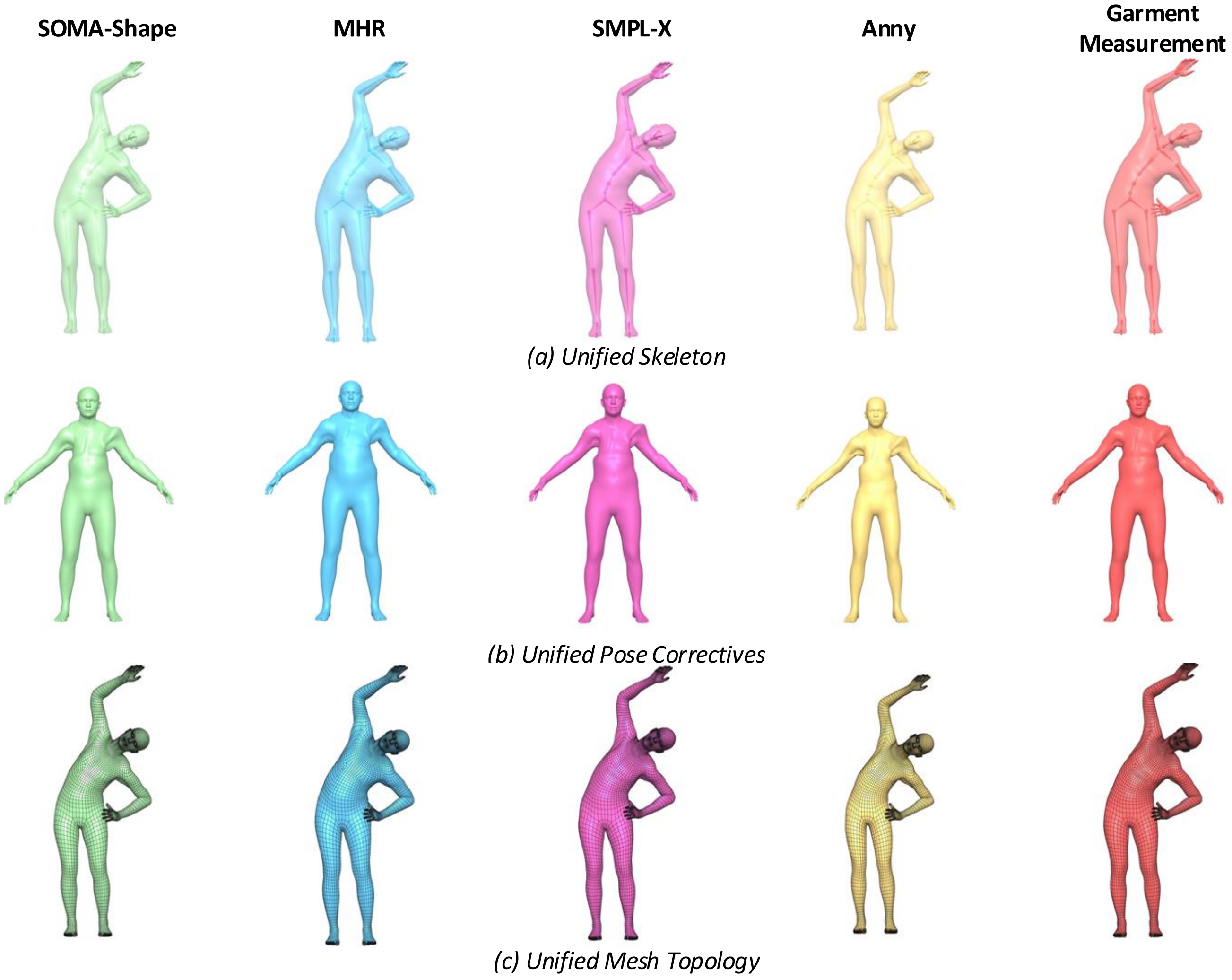
SOMA: Unifying Parametric Human Body Models

Jun Saito*, **Jiefeng Li***, **Michael de Ruyter***, **Miguel Guerrero**, **Edy Lim**, **Ehsan Hassani**, **Roger Blanco Ribera**, **Hyejin Moon**, **Magdalena Dadela**, **Marco Di Lucca**, **Qiao Wang**, **Xueting Li**, **Jan Kautz**, **Simon Yuen**, **Umar Iqbal***

NVIDIA

*Core Contributors

https://github.com/NVlabs/SOMA-X



(a) Unified Skeleton

(b) Unified Pose Correctives

(c) Unified Mesh Topology

Figure 1: SOMA unifies five heterogeneous parametric body models (SOMA-Shape, MHR, SMPL-X, Anny, and GarmentMeasurements) under a single animation pipeline. **(a) Unified Skeleton:** despite originating from entirely different identity spaces, joint hierarchies, and mesh resolutions, all five models are driven by the same SOMA skeleton in an identical pose, with no model-specific retargeting. **(b) Unified Pose Correctives:** a single MLP correctives model trained once on the shared SOMA topology produces anatomically plausible pose-dependent deformations for all backends, mitigating standard LBS artifacts without per-model corrective learning. **(c) Unified Mesh Topology:** all identity models share the same mesh structure, enabling skinning weights, deformation priors, and correctives to transfer seamlessly across backends.

Related projects that already support SOMA:

- **SOMA Retargeter** for SOMA to humanoid retargeting: https://github.com/NVIDIA/soma-retargeter
- **GEM** for video pose estimation: https://github.com/NVlabs/gem-x
- **Kimodo** for controllable text-to-motion generation: https://github.com/nv-tlabs/kimodo
- **BONES-SEED** is the largest human(oid) motion dataset (150k motions): https://huggingface.co/datasets/bones-studio/seed
- **ProtoMotions** is a simulation and learning framework for human(oids): https://github.com/NVlabs/ProtoMotions
- **GEAR SONIC** is a humanoid behavior foundation model: https://github.com/NVlabs/GR00T-WholeBodyControl

Abstract

Parametric human body models are foundational to human reconstruction, animation, and simulation, yet they remain mutually incompatible: SMPL, SMPL-X, MHR, Anny, and related models each diverge in mesh topology, skeletal structure, shape parameterization, and unit convention, making it impractical to exploit their complementary strengths within a single pipeline. We present SOMA, a unified body layer that bridges these heterogeneous representations through three abstraction layers. *Mesh topology abstraction* maps any source model's identity to a shared canonical mesh in constant time per vertex. *Skeletal abstraction* recovers a full set of identity-adapted joint transforms from any body shape, whether in rest pose or an arbitrary posed configuration, in a single closed-form pass, with no iterative optimization or per-model training. *Pose abstraction* inverts the skinning pipeline to recover unified skeleton rotations directly from posed vertices of any supported model, enabling heterogeneous motion datasets to be consumed without custom retargeting. Together, these layers reduce the $O(M^2)$ per-pair adapter problem to $O(M)$ single-backend connectors, letting practitioners freely mix identity sources and pose data at inference time. The entire pipeline is fully differentiable end-to-end and GPU-accelerated via NVIDIA-Warp.

1. Introduction

Parametric human body models are a cornerstone of computer vision, computer graphics, and physical AI, enabling reconstruction, animation, and simulation of human motion at scale. The most widely used family, SMPL (Loper et al., 2015; Pavlakos et al., 2019), defines meshes with compact linear shape spaces and has become the de facto target for pose estimation, motion generation, and avatar synthesis (Li et al., 2025; Shen et al., 2024; Wang et al., 2026; Zhang et al., 2024). MHR (Ferguson et al., 2025) introduces explicit bone-length parameterization that more faithfully captures skeletal diversity across people, addressing a well-known limitation of PCA-only shape models. Anny (Brégier et al., 2025) constructs its shape space from anthropometric measurements rather than 3D scans, providing semantic control over age, height, weight, and body composition that spans the full human lifespan from infants to elders, aiming to remove the demographic biases inherent in scan-collected data. GarmentMeasurements (Korosteleva and Sorkine-Hornung, 2023) extends shape representation to clothing-aware body proportions encoded via body measurements.

Despite this diversity of available models, a concrete fragmentation problem persists. Each model defines its own mesh topology, joint hierarchy, unit convention, and parameter space. A practitioner wishing to combine Anny's interpretable phenotype control with an SMPL-compatible motion capture dataset must implement separate topology-transfer pipelines, independent skeleton-fitting routines, and bespoke coordinate-frame conversions for every model pair. Supporting M models naively requires $O(M^2)$ per-pair adapters; in practice, this forces early model commitment and forfeits the complementary strengths of alternatives. No unified interface currently exists that lets a researcher freely mix identity source and pose parameterization.

We introduce SOMA, a canonical body topology and rig that serves as a universal pivot for heterogeneous parametric body models (see Fig. 2). Rather than replacing existing models, SOMA maps their rest-shape outputs to a single shared representation, after which any identity model can be animated through one unified LBS pipeline. This reduces the $O(M^2)$ adapter problem to $O(M)$ single-backend connectors, each implemented once and composed freely at inference time.

Our contributions are fourfold:

1. **Identity-Pose Decoupling via a Canonical Topology (`SOMALayer`).** We propose a framework that maps the rest-shape output of any supported parametric model to a canonical SOMA mesh and rig, explicitly separating identity representation from kinematic parameterization. A single pose interface drives any identity source without model-specific adaptation code at runtime, and a unified pose-dependent

correctives model generalizes to all backends.

2. **Mesh Topology Abstraction.** We introduce a topology abstraction module that pre-computes a fixed 3D barycentric correspondence between each source model's neutral mesh and the SOMA canonical mesh at initialization. At runtime, identity transfer requires no neural forward pass and no iterative solver.
3. **Skeletal Abstraction.** We present a backend-agnostic skeleton fitting algorithm that exploits the shared mesh correspondence to precisely fit any template skeleton into a new body shape. Given the transferred rest shape, it recovers identity-adapted world-space joint transforms in a single analytical forward pass with no iterative optimization or per-model training.
4. **Pose Abstraction.** We introduce a pose abstraction module that recovers SOMA skeleton rotations from posed vertices of any supported model via analytical inverse-LBS with Newton-Schulz orthogonalization, enabling direct conversion of motion data from SMPL, MHR, and other models into SOMA's unified skeleton convention.

The entire SOMA forward pass is fully differentiable end-to-end, making it directly usable as a differentiable layer in large-scale optimization and ML training pipelines.

2. Related Work

2.1. Parametric Body Models

The field has seen significant evolution in parametric human modeling (Anguelov et al., 2005; Brégier et al., 2025; Ferguson et al., 2025; Loper et al., 2015; Osman et al., 2020; Pishchulin et al., 2017; Xu et al., 2020). **SMPL** (Loper et al., 2015) introduced a vertex-based linear blend skinning model with learned corrective blend shapes, which became the de facto standard with 6,890 mesh vertices and a compact PCA shape space. **STAR** (Osman et al., 2020) proposed sparser skinning weights to reduce undesirable cross-joint coupling. SMPL-H (Romero et al., 2017) and **SMPL-X** (Pavlakos et al., 2019) extended the SMPL family with fully articulated hands, via **MANO** (Romero et al., 2017), and an expressive face, respectively. **MHR** (Ferguson et al., 2025) addresses skeletal ambiguity by explicitly modeling bone lengths to improve fitting accuracy across body proportions. **Anny** (Brégier et al., 2025) builds its shape space from anthropometric measurements rather than 3D scans, enabling phenotype control (age, height, weight) that spans infants to elders. Each of these models defines its own mesh topology, joint hierarchy, and parameter space. SOMA does not replace any of them; instead, it provides a canonical mesh topology and rig that any supported backend can drive through a single unified pipeline.

2.2. Human Motion Estimation and Generation

A rich body of work estimates 3D human pose and shape from monocular images (Goel et al., 2023; Iqbal et al., 2021; Kanazawa et al., 2018; Kocabas et al., 2021, 2024; Kolotouros et al., 2019; Patel and Black, 2025; Sárándi and Pons-Moll, 2024; Wang et al., 2025; Yuan et al., 2022), videos (Choi et al., 2021; Goel et al., 2023; Kocabas et al., 2020; Shen et al., 2024; Shin et al., 2023; Wang et al., 2026), and generates motion from diverse conditioning signals such as text, music, and scene context (Li et al., 2025; Petrovich et al., 2024; Tevet et al., 2023; Yuan et al., 2023; Zhang et al., 2022, 2024). The vast majority of these systems are built around SMPL or SMPL-X as the target representation; more recently, methods such as MultiHMR (Baradel et al., 2024), Sam-3D-Body (Yang et al., 2026), and DuoMo (Wang et al., 2026) have started to adopt MHR and Anny to better capture bone-length and age-range diversity. However, whether estimating or generating, all of these systems are trained to output parameters for one specific body model and must be retrained whenever the target representation changes. SOMA decouples identity model selection from the estimation pipeline: a pose estimator or generative model outputting SOMA-compatible joint parameters can drive any supported identity, SMPL, MHR, Anny, or others, at inference time without retraining, and can be supervised with body shape labels from all backends simultaneously.

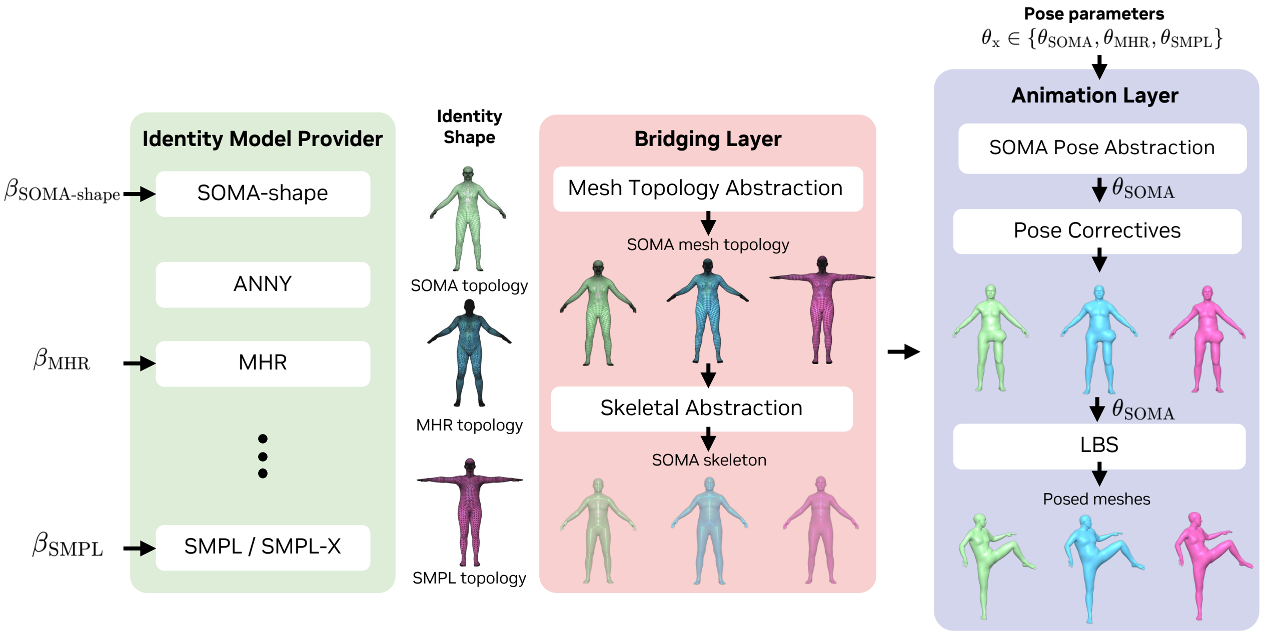


Figure 2: **Overview of SOMA.** SOMA decouples body identity from pose through three sequential layers. *Identity Model Provider* (left): any supported backend (SOMA-shape, Anny, MHR, SMPL/SMPL-X, or GarmentMeasurements) maps its own shape parameters β_s to a rest-shape mesh in its native topology. *Bridging Layer* (middle): two abstraction steps canonicalize the source identity into a unified representation. Mesh Topology Abstraction transfers the rest shape to the shared SOMA topology via pre-computed barycentric coordinates; Skeletal Abstraction then fits the shared J=77-joint SOMA rig to the transferred rest shape in a single closed-form pass, with no iterative optimization or per-identity training. *Animation Layer* (right): all identity models are animated through the shared SOMA skeleton using θ_{SOMA} joint rotations. When motion data arrives in another model's convention, *i.e.* $\theta_{\mathrm{x}} \in \{\theta_{\mathrm{MHR}}, \theta_{\mathrm{SMPL}}, \ldots\}$, Pose Abstraction converts it to θ_{SOMA} by analytically inverting the LBS pipeline; this step is bypassed when pose is already in the SOMA convention. A shared MLP Pose Correctives model then predicts pose-dependent vertex displacements to correct LBS artifacts, and Linear Blend Skinning produces the final posed mesh. The entire pipeline is fully differentiable end-to-end.

3. Method

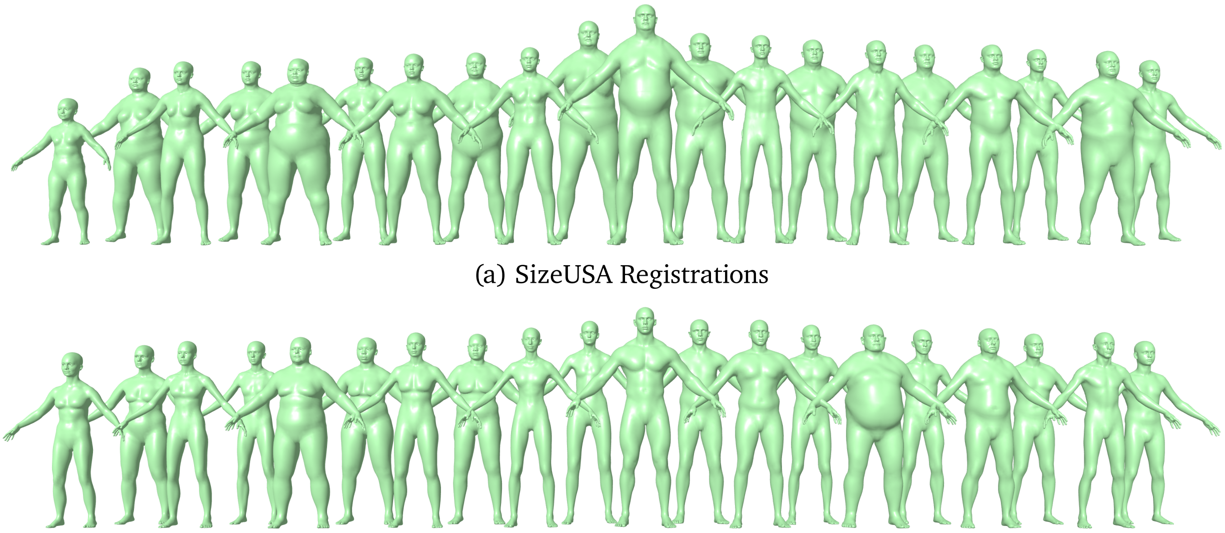
SOMA is a modular framework for unified parametric body modeling. Its core runtime component, `SOMALayer`, accepts shape parameters from any supported identity backend alongside pose parameters, and produces posed mesh vertices and joint positions in meters. Fig. 2 illustrates the full pipeline.

3.1. Overview and Notation

Let $V_h \in \mathbb{R}^{N_h \times 3}$ denote the SOMA canonical mesh with N_h vertices, F_h its triangle faces, and $J = 77$ its joint count (excluding the root). For each supported backend $s \in \{$NOVA, MHR, Anny, SMPL, SMPL-X, Garment$\}$, let $\mathcal{M}_s(\beta_s)$ denote the backend's rest-shape generator, which maps identity parameters β_s to a source mesh $V_s \in \mathbb{R}^{N_s \times 3}$ in the backend's native unit. SOMA's forward pass transforms any (V_s, θ) pair into posed SOMA mesh vertices via three sequential steps: (1) mesh topology abstraction; (2) closed-form skeleton fitting; and (3) LBS posing. Every step is fully differentiable, so the entire pipeline can serve as a differentiable layer in optimization and learning frameworks.

3.2. Identity Model Provider

The identity model provider takes the native shape parameters β_s of any supported backend and maps them to a rest-shape mesh in that backend's native topology. SOMA integrates five interchangeable backends, each with its own strengths, allowing users to easily adopt the identity model of their preference within a single



(a) SizeUSA Registrations

(b) Triplegangers Registrations

Figure 3: **Training data for the SOMA-Shape PCA model.** (a) A subset of the 9,326 SizeUSA body scans registered to the SOMA topology, exhibiting a wide range of body weights and proportions. (b) A subset of the 303 Triplegangers scans, registered to the same topology. All meshes are reposed to a canonical A-pose before PCA fitting, and mirror-augmented across the sagittal plane to enforce bilateral symmetry.

animation framework.

SOMA-Shape. SOMA's own shape space uses $K{=}128$ PCA components trained on 9,326 SizeUSA body scans ([TC]^2, 2004), 303 Triplegangers photogrammetry scans (Triplegangers, 2025), and samples distilled from the GarmentMeasurements PCA model (Korosteleva and Sorkine-Hornung, 2023), with a 40/40/20 mixing ratio learned by incremental PCA (Ross et al., 2008). We show example registrations in Fig. 3.

SMPL / SMPL-X (Loper et al., 2015; Pavlakos et al., 2019) parameterize body shape via PCA blend shapes (10 components for SMPL, 300 for SMPL-X) learned from registered 3D body scans and dominates research adoption.

MHR (Ferguson et al., 2025) parameterizes body shape via a combination of PCA identity coefficients and explicit bone-length scale factors that directly modulate skeletal proportions, providing fine-grained control over body surface shape and skeletal proportions.

Anny (Brégier et al., 2025) parameterizes body shape using six anthropometric phenotypes (gender, age, muscle, weight, height, proportions) that drive multi-linear blendshapes spanning infants to elders. Anny is the sole backend capable of representing subjects below 18 years of age, hence making it the natural choice for applications requiring age-diverse or child-inclusive digital humans.

GarmentMeasurements (Korosteleva and Sorkine-Hornung, 2023) encodes body shape via 15 PCA components fitted to body scan data, similar to SMPL/SMPL-X, and is often adopted for garment modeling literature.

3.3. Mesh Topology Abstraction

The mesh topology abstraction layer maps diverse source topologies from the identity model provider to the SOMA canonical mesh. We pre-compute a fixed 3D barycentric correspondence at initialization and apply it as a lightweight gather at runtime, as illustrated in Fig. 4 illustrates.

More specifically, given the source neutral mesh (V_s, F_s) and a SOMA wrap mesh $V_h^{(s)}$ (a version of the SOMA template manually registered to the source model's neutral pose by an artist), we compute for each SOMA

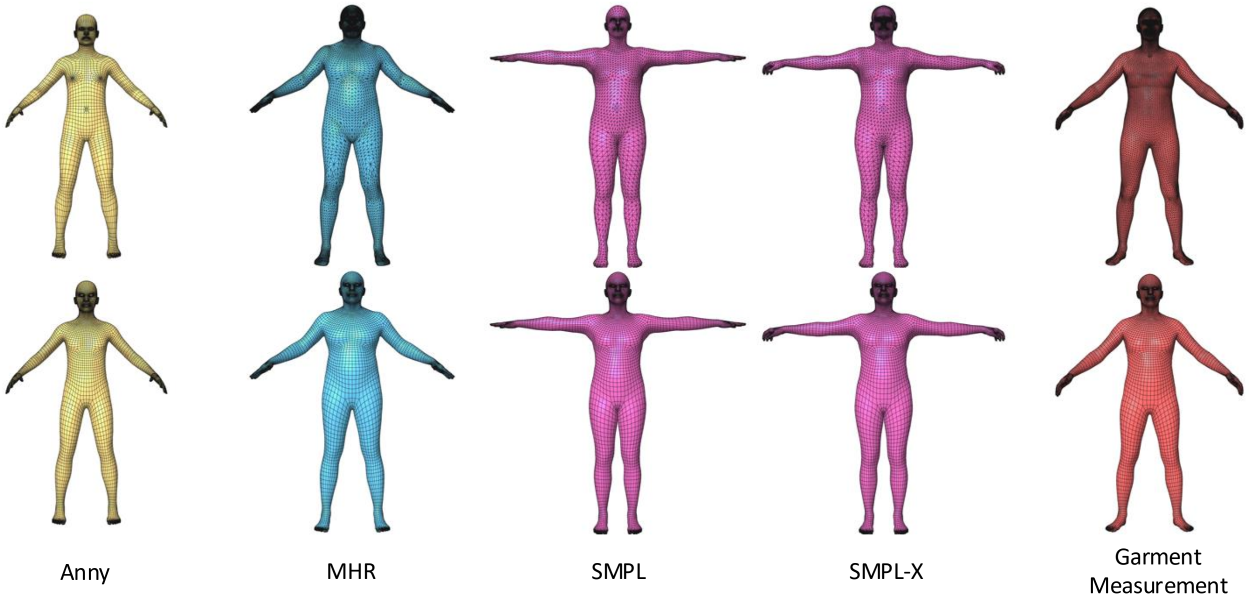


Figure 4: **Mesh topology abstraction.** Top: native mesh topologies of each identity model. Bottom: the same identities mapped to the shared SOMA topology via 3D barycentric interpolation. This common mesh serves as the pivot for all cross-model operations—skeleton fitting, pose transfer, correctives, and shape-space comparison all operate on a single canonical topology regardless of the source model.

vertex $\mathbf{v}_i^h$ its 3D barycentric coordinates within a local tetrahedron of the source mesh. For the closest source triangle $f_j = (u_1, u_2, u_3)$ to $\mathbf{v}_i^h$, we lift it to a tetrahedron by adding a fourth vertex along the face normal, $u_4 = u_1 + (u_2 - u_1) \times (u_3 - u_1)$, and solve for the barycentric coordinates $\mathbf{b} \in \mathbb{R}^4$ via a $3{\times}3$ linear system. This tetrahedral lifting handles query points slightly off the surface without degeneracy. Unlike 2D barycentric projection, 3D tetrahedral interpolation preserves volume in regions without clear surface correspondence, for example, when mapping between models with and without individual toes. The pre-computation runs once at initialization; its output-, a face-index array $\mathbf{f} \in \mathbb{Z}^{N_h}$ and coordinate array $\mathbf{B} \in \mathbb{R}^{N_h \times 4}$, is stored as a fixed buffer.

At runtime, given deformed source vertices $V_s(\beta) \in \mathbb{R}^{N_s \times 3}$, each SOMA vertex is reconstructed as a weighted combination of its corresponding tetrahedron's vertices:

$$\mathbf{v}_i^h(\beta) = \sum_{k=0}^{3} B_{ik} \cdot \tilde{V}_s(\beta)[\,\mathbf{F}^{\text{tet}}_{\mathbf{f}_i,\,k}\,], \tag{1}$$

where $\tilde{V}_s(\beta)$ is the source mesh augmented with one normal-offset point per face. The cost is independent of the source vertex count N_s.

3.4. Skeletal Abstraction

Once all source meshes share a common topology (Sec. 3.3), we need a single skeletal structure to drive their pose. We introduce `SkeletonTransfer`, a backend-agnostic algorithm that precisely fits any template skeleton into a new body shape given only the shared mesh correspondence. In SOMA, we apply it to fit a $J{=}77$ joint rig; given a rest shape $V_h(\beta) \in \mathbb{R}^{N_h \times 3}$ on the SOMA topology, it recovers the full set of world-space joint transforms $\{T_k\}_{k=1}^{J} \subset SE(3)$ in two analytical stages: joint position regression and joint rotation fitting. Fig. 5 illustrates how the fitted skeleton adapts to bodies of varying proportions.

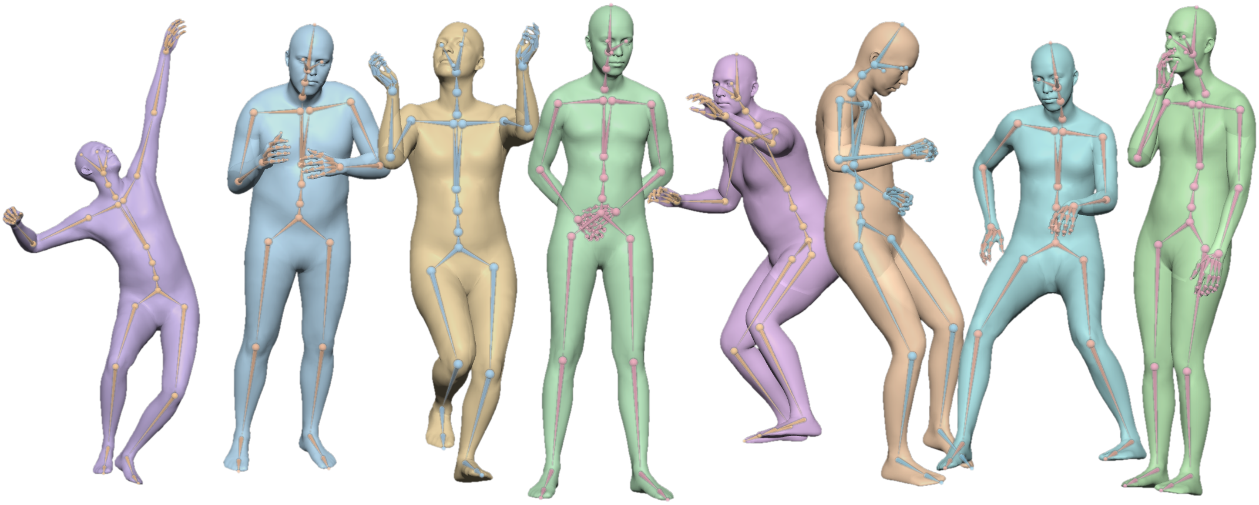


Figure 5: **Skeleton fitting on posed SAM 3D Body identities.** Eight MHR identities in diverse poses with the SOMA skeleton fitted via `SkeletonTransfer` (Sec. 3.4). Unlike joint regressors that assume a rest pose, our method generalizes to arbitrary posed shapes: joint positions are regressed via RBF interpolation, and joint rotations are recovered by Procrustes alignment, both in a single analytical forward pass with no iterative optimization.

3.4.1. Stage 1: Joint position regression via RBF

For each joint $k \in \{1, \ldots, J\}$, we pre-build a per-joint Radial Basis Function regressor from the canonical bind-pose mesh. The regressor uses the subset of vertices $\mathcal{N}_k$ that have non-zero skinning weight for joint k or its parent. Given the canonical bind-shape vertex positions V^{bind}, the RBF basis weights $\mathbf{w}_k$ are solved once by the linear system:

$$\Phi(\mathcal{N}_k)\, \mathbf{w}_k = \mathbf{j}_k^{\text{bind}}, \tag{2}$$

where Φ is the linear RBF kernel evaluated at the neighborhood vertex positions, and $\mathbf{j}_k^{\text{bind}}$ is the canonical joint position.

At runtime, given identity rest shape $V_h(\beta)$, joint k's world-space position is predicted as:

$$\mathbf{j}_k(\beta) = \Phi\big(V_h(\beta)_{\mathcal{N}_k}\big)\, \mathbf{w}_k, \tag{3}$$

which is a single linear operation. All joint positions are computed in parallel via a pre-assembled sparse matrix $\mathbf{W}_{\text{RBF}} \in \mathbb{R}^{J \times N_h}$, reducing the full joint position update to one sparse matrix multiplication:

$$J(\beta) = \mathbf{W}_{\text{RBF}}\, V_h(\beta)^T. \tag{4}$$

3.4.2. Stage 2: Joint rotation fitting via Kabsch alignment

Joint positions alone do not fully define the skeleton since each joint also requires an orientation that establishes its local coordinate frame. Since source models assume different canonical poses (*e.g.*, T-pose vs. A-pose), these orientations cannot be copied from the bind pose and must be fitted to the identity's rest shape. With identity-adapted joint positions $\{\mathbf{j}_k(\beta)\}$ in hand, we recover the rotation component of each world-space joint transform via a two-step Kabsch alignment procedure.

Stage 2a: Inverse LBS initialization. For joint k, let $\mathcal{V}_k$ be the set of vertices with non-negligible skinning weight for joint k. We estimate an initial global rotation $R_k^{\text{init}} \in SO(3)$ by solving the weighted orthogonal Procrustes problem (Kabsch, 1976):

$$R_k^{\text{init}} = \underset{R \in SO(3)}{\arg\min} \sum_{\mathbf{v} \in \mathcal{V}_k} \left\| R\big(\mathbf{v}^{\text{bind}} - \mathbf{j}_k^{\text{bind}}\big) - \big(\mathbf{v}(\beta) - \mathbf{j}_k(\beta)\big) \right\|^2. \tag{5}$$

This is solved via SVD of the cross-covariance matrix.

Stage 2b: Child bone alignment. The initial rotation R_k^{init} aligns the skinned vertex cloud but may not correctly orient the bone vectors toward child joints. We compute a refinement rotation R_k^{align} that aligns the rotated bind bone vectors $R_k^{\text{init}}(\mathbf{j}_c^{\text{bind}} - \mathbf{j}_k^{\text{bind}})$ to the target bone vectors $\mathbf{j}_c(\beta) - \mathbf{j}_k(\beta)$. For joints with a single child (the majority of the skeleton), this is the shortest-arc (Rodrigues) rotation between two vectors; for joints with multiple children, we solve the Procrustes problem (Eq. (5)) over the set of child bone vectors. The final world-space rotation is $R_k = R_k^{\text{align}} \cdot R_k^{\text{init}} \cdot R_k^{\text{bind}}$, where R_k^{bind} is the canonical bind-pose world rotation, and the complete transform is $T_k = \mathrm{SE3}(R_k, \mathbf{j}_k(\beta))$. Both paths are fully vectorized across all joints via NVIDIA Warp custom kernels, enabling massively parallel GPU execution with no sequential joint loop.

3.5. Animation Layer

Given the identity-adapted joint transforms $\{T_k(\beta)\}$ and rest shape $V_h(\beta)$, SOMA's animation layer applies standard Linear Blend Skinning (LBS) and correctives to produce animated vertices.

3.5.1. Posing

Given input pose parameters (axis-angle $(B, 77, 3)$ or rotation matrices $(B, 77, 3, 3)$), together with an optional root translation $t_0 \in \mathbb{R}^3$, forward kinematics computes global joint transforms $\{G_k(\theta)\}_{k=0}^{J-1}$ by composing local rotations up the joint hierarchy. SOMA can optionally apply *joint orient* (pose-relative parameterization) where input rotations are expressed relative to the joint's canonical pose (usually T-pose or A-pose), which matches the convention of many parametric human models and their associated datasets. Posed vertex positions are then:

$$\mathbf{v}_i' = \sum_{k=0}^{J-1} w_{ik}\, G_k(\theta)\, T_k^{\text{bind}\,-1}\, \tilde{\mathbf{v}}_i, \tag{6}$$

where $\tilde{\mathbf{v}}_i$ is the homogeneous rest-shape position and w_{ik} is the skinning weight.

3.5.2. Pose-Dependent Correctives

Standard LBS produces well-known artifacts at joints undergoing large rotations (elbow, shoulder, knee). Some identity models ship with their own correctives (e.g. MHR), while others such as Anny do not. SOMA provides a single unified correctives model that applies to *all* backends, by operating on the shared canonical topology and rest pose established by the preceding abstraction layers.

The correctives are predicted by a lightweight non-linear MLP network and applied to the rest shape before skinning:

$$V_h^{\text{corr}}(\beta, \theta) = V_h(\beta) + f_{\text{MLP}}(\theta), \tag{7}$$

where f_{MLP} takes as input the local joint rotations $\{R_k(\theta)\}_{k=0}^{J-1}$ in 6D representation (Zhou et al., 2019).

The MLP follows a two-stage structure inspired by MHR (Ferguson et al., 2025): joint rotations are mapped to a bank of $K = J \times C$ corrective activations ($C = 24$), which are then mapped to per-vertex displacements. Fixed anatomical masks derived from skinning weights and geodesic distances enforce spatial locality and sparsity. Training data is distilled from MHR by sampling $\approx$80,000 MHR posed meshes onto the SOMA topology via barycentric interpolation (Sec. 3.3) and pose inversion (Sec. 3.6)—a large-scale distillation made practical by SOMA's unified topology and pose abstraction. Fig. 6 shows some examples of our unified correctives for all body models.

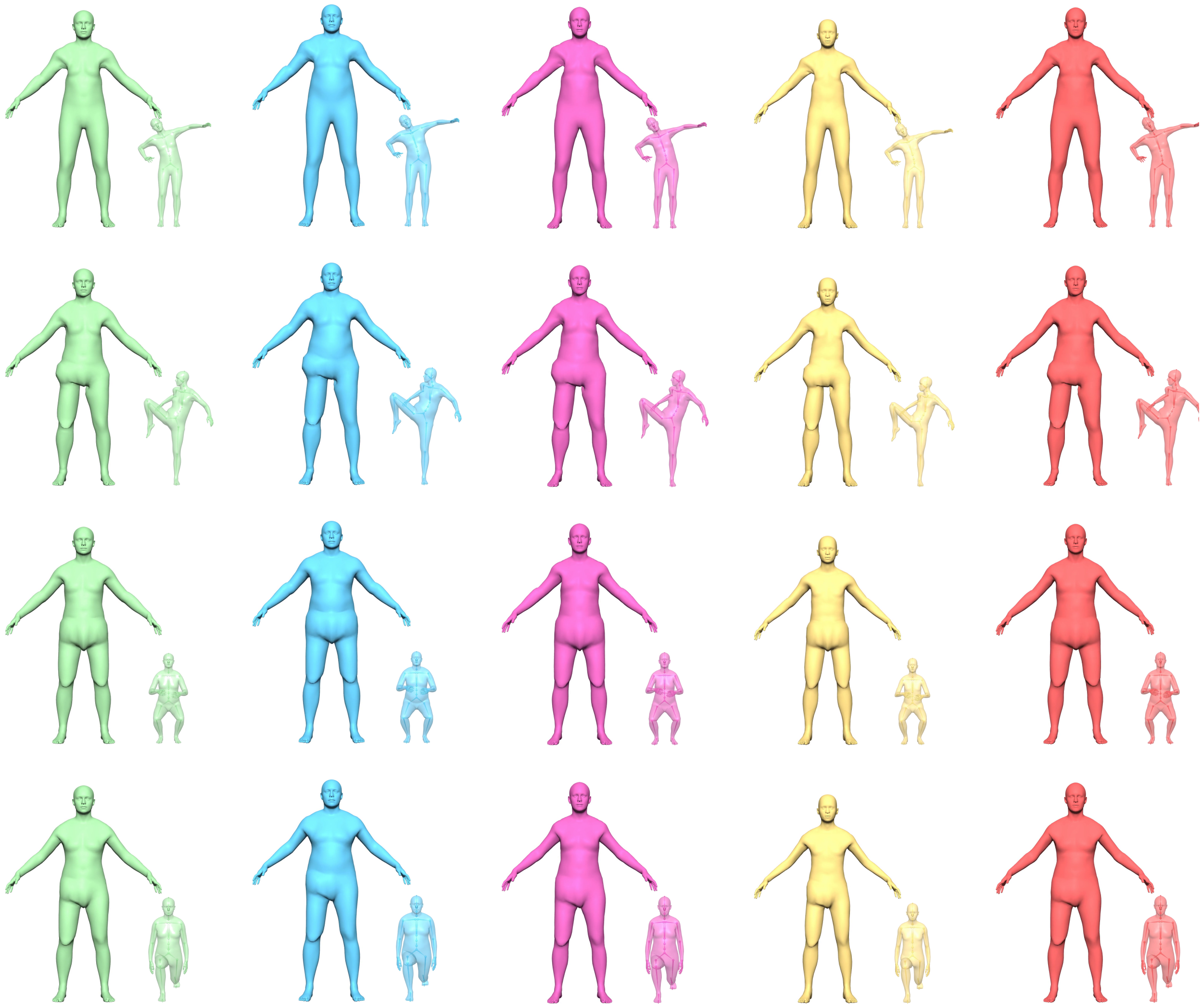


Figure 6: **Unified pose correctives across identity backends.** Each row shows a different pose; columns correspond to SOMA-Shape, MHR, SMPL, Anny, and GarmentMeasurements. For each cell, the left mesh shows the corrective displacement applied to the canonical rest shape, and the right mesh shows the final posed result. A single correctives model trained once on SOMA's canonical topology produces anatomically plausible deformations for all backends.

3.6. Pose Abstraction

The sections above describe the forward path of SOMA: given an identity and a pose, the framework produces a posed mesh in a unified representation. A complementary operation is equally important in practice—recovering SOMA pose parameters *from* an already-posed mesh. We call this *pose abstraction*: just as topology abstraction (Sec. 3.3) and skeletal abstraction (Sec. 3.4) unify heterogeneous body shapes into a single identity representation, pose abstraction unifies heterogeneous pose data into a single SOMA skeleton convention. This enables motion sequences captured or generated with SMPL, MHR, Anny, or any other supported backend to be directly consumed by downstream SOMA applications without custom retargeting. Large-scale motion datasets such as AMASS (Mahmood et al., 2019) and SAM 3D Body (Yang et al., 2026) thereby become natively usable through SOMA.

The core algorithmic challenge is *pose inversion*: recovering per-joint rotations from posed vertex positions—the inverse of the forward kinematic and LBS pipeline described in Sec. 3.5. We describe the pose inversion algorithm below.

Multi-topology input. Posed vertices from any supported mesh topology—not only the native SOMA mesh—are accepted as input. When the input topology differs from SOMA's canonical mesh, the same barycentric transfer used in the forward path (Sec. 3.3) first maps the input vertices to the SOMA topology. From this point onward, the inversion algorithm operates entirely in SOMA's canonical vertex and skeleton space.

Initialization via skeleton transfer. Pose inversion begins with the same Kabsch-based skeleton fitting procedure described in Sec. 3.4: a single-pass RBF joint regression followed by Procrustes alignment provides an initial world-space rotation estimate for each joint. This initialization is already a reasonable approximation of the target pose and can serve as a standalone fast solver when only coarse pose recovery is needed.

Iterative inverse-LBS refinement. Starting from the skeleton-transfer initialization, the algorithm refines joint rotations level-by-level in parent-to-child order. For joint k, the skinned vertex positions are decomposed into a subtree contribution (vertices predominantly influenced by k and its descendants) and a non-subtree contribution from already-solved ancestor joints. The ancestor contribution is subtracted from the observed posed vertices, isolating the local deformation attributable to joint k alone. A Procrustes alignment (Eq. (5)) is then solved for the local rotation.

Newton-Schulz orthogonalization. The standard Kabsch algorithm computes the nearest rotation matrix from the cross-covariance matrix $H = A^T B$ via SVD: $R = UV^T$ where $H = U\Sigma V^T$. However, when the point cloud contributing to a joint's covariance is near-coplanar—as commonly occurs at body parts such as clavicles—the smallest singular value σ_3 approaches zero and the corresponding singular vector becomes ill-defined. Under these conditions, small perturbations in the input vertices can flip the sign of a singular vector between consecutive frames, causing a discontinuous 180 rotation jump ("shoulder popping"). To avoid this instability, our iterative refinement replaces SVD with Newton-Schulz orthogonalization (Kovarik, 1970), which computes the polar factor of H via the fixed-point iteration

$$R_{i+1} = \tfrac{1}{2} R_i (3I - R_i^T R_i), \quad R_0 = H / \|H\|_\infty, \tag{8}$$

where $\|H\|_\infty$ is the infinity norm (maximum absolute row sum) that guarantees convergence. Because this iteration refines the rotation estimate *continuously* from its current value rather than decomposing and reassembling singular vectors, it is immune to the sign-flipping discontinuity inherent in SVD for near-degenerate covariance matrices.

Hierarchical scheduling. The refinement schedule mirrors the skeletal hierarchy: body joints are solved first, followed by optional finger joint refinement, and a final global pass covers all joints simultaneously. This coarse-to-fine schedule ensures that large-scale body motion is resolved before fine-grained finger articulation, which would otherwise be contaminated by uncorrected upstream errors.

Optional autograd refinement. For applications that require higher accuracy at the cost of throughput, an optional gradient-based refinement stage is provided. Joint rotations are parameterized as continuous 6D vectors (Zhou et al., 2019) and optimized with Adam by backpropagating through the full FK+LBS computation (Eq. (6)). This stage must be warm-started from the analytical result: the FK+LBS objective is highly non-convex, and naïve optimization from the bind pose fails to converge, settling into a local minimum with entirely incorrect limb placement (Sec. 4.2). With the analytical initialization, autograd refinement converges rapidly and can further reduce error at extremities (hands, feet, head) by optimizing through the full kinematic chain with per-vertex loss weighting. The analytical solver alone achieves approximately 1,200 frames per second on an NVIDIA RTX 5000 Ada GPU, while the autograd path runs at 16–18 FPS for 100 optimizer steps, making each mode suitable for different points on the speed–accuracy tradeoff curve.

Table 1: **Topology abstraction fidelity across backends.** Closest-point-to-mesh distance (mm) between the SOMA rest shape and the native source mesh, measured over 100 diverse identities per backend. "Wrap" is the baseline registration error of the pre-registered SOMA wrap mesh against the source neutral mesh. Vertices in facial inner geometry (eye bags, mouth bag) and between-toes regions—which have no correspondence in most source topologies—are excluded.

Backend	Src. Verts	Mean (mm)	Std (mm)	P95 (mm)	Wrap Mean	Wrap P95
SOMA native	–	0.0	0.0	0.0	–	–
SMPL	6,890	0.12	0.45	0.71	0.12	0.74
SMPL-X	10,475	0.06	0.22	0.45	0.06	0.47
Anny	13,718	0.01	0.12	0.01	0.01	0.01
MHR	$\sim$18k	0.40	0.73	1.49	0.31	1.28

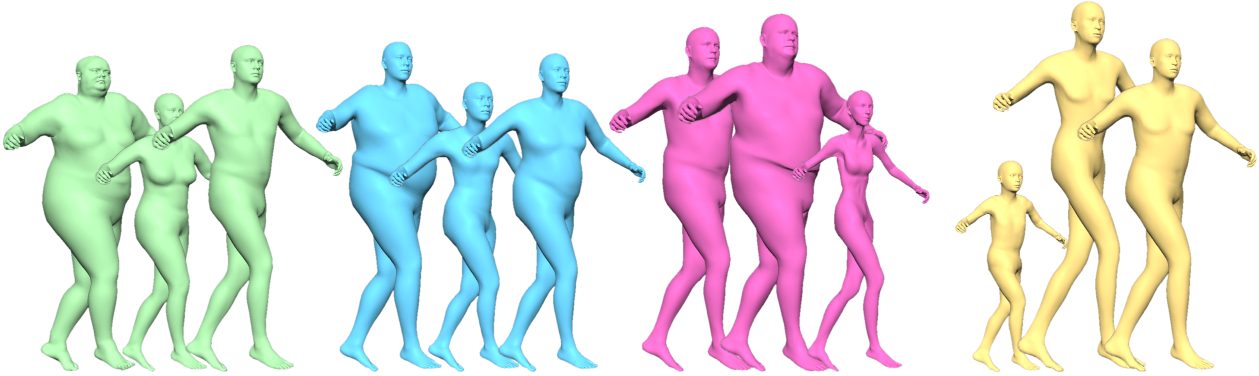


Figure 7: **Shape diversity across backends driven by a single pose.** Three sampled identities per backend—SOMA-Shape (green), MHR (blue), SMPL (pink), and Anny (yellow)—all driven by the same skeletal pose. Despite originating from entirely different generative models, all bodies share the same pose interpretation, illustrating the plug-and-play nature of SOMA identity-pose decoupling.

4. Evaluation

We evaluate SOMA along four dimensions: topology abstraction fidelity (Sec. 4.1), pose inversion accuracy (Sec. 4.2), runtime performance (Sec. 4.3), and cross-model shape-space comparison (Sec. 4.4).

4.1. Topology Abstraction Fidelity

The topology transfer is the first stage of the pipeline and any error here propagates to all downstream tasks. Tab. 1 reports per-vertex transfer errors for each backend over 100 diverse identities. For each SOMA vertex, we query the closest point on the native source mesh surface and report the L_2 distance. This measures the geometric information loss introduced by the barycentric interpolation, without conflating it with unit convention or alignment artifacts.

The SOMA-native backend incurs zero error by construction. SMPL and SMPL-X achieve sub-millimeter mean errors (0.12 mm and 0.06 mm respectively), and their transfer errors closely match the wrap baseline, confirming that the barycentric interpolation introduces negligible additional distortion beyond the one-time mesh registration. Anny achieves near-zero error (0.01 mm mean), reflecting a particularly clean wrap registration. MHR shows a slightly higher mean error (0.40 mm), reflecting its denser mesh and more complex geometry, but remains well below 1 mm; the modest increase over the wrap baseline (0.31 mm) indicates that the topology transfer generalizes well across the MHR shape space. All P95 errors stay below 1.5 mm.

Fig. 7 shows the shape diversity achieved across all backends using identical pose parameters, demonstrating that the unified pipeline faithfully represents the shape characteristics of each source model.

Table 2: **Pose inversion accuracy and throughput on AMASS.** Per-vertex reconstruction error (mm) and throughput (frames/sec) on an NVIDIA A100 GPU. "Skel. transfer" is the raw skeleton-fitting initialization with no iterative refinement. "Analytical" adds inverse-LBS refinement with Newton-Schulz orthogonalization (body=2, full=1). "Autograd FK" optimizes 6D rotation parameters through FK+LBS with Adam (100 iterations); "no init" starts from the bind pose, "w/ init" warm-starts from the skeleton transfer result. "Analytical + Autograd" warm-starts autograd from the analytical result (10 iterations).

Method	Mean (mm)	Median (mm)	Max (mm)	FPS
Skel. transfer only	16.5	13.9	80.1	17,393
Analytical	5.3	3.2	88.1	882
Autograd FK (no init)	501.8	479.4	1354.2	79
Autograd FK (w/ init)	4.1	2.1	81.5	78
Analytical + Autograd (10)	7.8	6.4	88.6	435

Table 3: **Per-region pose inversion error (MHR, 200 SAM 3D Body frames).** Mean per-vertex L_2 error (mm) by body region. "Analytical" = body=2, full=1. "+ Autograd" adds 100 Adam iterations warm-started from the analytical result.

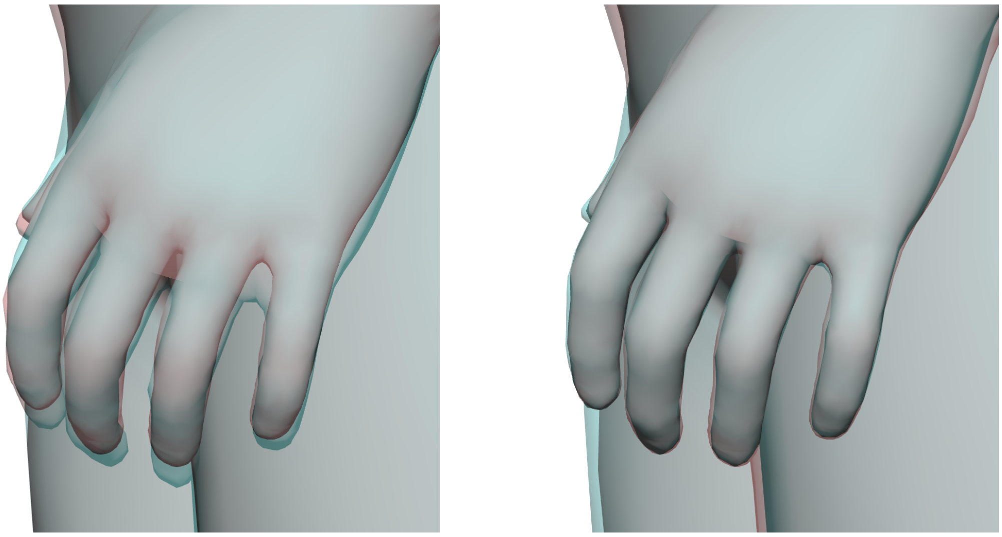
Method	All	Body	Hands	Feet	Head
Analytical	8.8	16.8	4.7	8.2	6.9
+ Autograd (100)	6.6	15.3	2.0	5.8	4.8
Reduction	25%	9%	**57%**	29%	30%

4.2. Pose Inversion Accuracy

Posed meshes are the common interface across heterogeneous body models, making mesh-to-joint-angle inversion the key operation for pose abstraction. Tab. 2 reports pose inversion accuracy and throughput for both solvers described in Sec. 3.6. We evaluate on the full AMASS dataset (Mahmood et al., 2019) (344 subjects, 2,265 motions, 40.3 hours, 19.8M frames). Errors are per-vertex L_2 distances between the original SMPL-X posed mesh and the SOMA reconstruction driven by the recovered rotations.

Fig. 9 qualitatively compares the three stages on SMPL and MHR backends. The skeleton transfer initialization alone (Sec. 3.4) provides a coarse but fast pose estimate (16.5 mm mean at 17,393 FPS), suitable for applications where speed dominates accuracy requirements. The analytical solver refines this to 5.3 mm mean error at 882 FPS via iterative inverse-LBS with Newton-Schulz orthogonalization. The autograd FK solver reaches 4.1 mm mean error by optimizing through the full FK+LBS chain, but *only when warm-started* from the skeleton transfer initialization; without initialization (starting from the bind pose), 100 Adam iterations fail to converge (501.8 mm mean error), demonstrating that the initialization is critical. The two solvers are complementary. The analytical path is fast and produces a near-optimal result in terms of global L_2 error across all vertices. The autograd FK path, by contrast, optimizes through the full FK+LBS chain and supports per-vertex loss weighting on extremities (hands, feet, head), giving explicit control over where the solver concentrates its effort. Tab. 3 breaks down per-vertex errors by body region on 200 SAM 3D Body (Yang et al., 2026) frames (MHR backend). Autograd FK refinement (100 iterations, warm-started from the analytical result) reduces hand error by 57% ($4.7 \to 2.0$ mm), foot error by 29% ($8.2 \to 5.8$ mm), and head error by 30% ($6.9 \to 4.8$ mm), while body trunk error decreases slightly ($16.8 \to 15.3$ mm)—the optimizer redistributes error away from the extremities and onto the trunk, which has more vertices to absorb it. Fig. 8 visualizes this trade-off on a hand close-up: the autograd result achieves a tight overlay at the fingertips, at the cost of a slightly increased offset on the body visible in the background.

Effect of Newton-Schulz orthogonalization. Fig. 10(a, b) compares the analytical solver using standard SVD-based Kabsch alignment against the Newton-Schulz variant described in Sec. 3.6. The crops are taken



(a) Analytical only

(b) Analytical + Autograd FK

Figure 8: **Hand zoom: analytical vs. autograd FK refinement.** MHR backend on SAM 3D Body (teal = ground truth, red = SOMA reconstruction). (a) The analytical solver is near-optimal globally but leaves residual misalignment at the fingertips. (b) Adding 100 autograd FK iterations redistributes error toward the body trunk (visible as a slightly increased offset in the background), achieving a tight overlay at the fingers.

from the shoulder region of the same SMPL frame, where the contributing vertex cloud is near-coplanar. With SVD, the near-zero third singular value causes a sign flip in the rotation solution, producing a visible discontinuous offset at both shoulders ("shoulder popping"). Newton-Schulz orthogonalization avoids this instability by iteratively refining the rotation estimate without decomposing singular vectors, yielding a smooth and accurate result. To quantify the temporal effect, we measure frame-to-frame shoulder-region vertex error change across the full 1,606-frame sequence: SVD exhibits a peak error oscillation of 1.6 mm/frame at the shoulders, compared to 0.8 mm/frame for Newton-Schulz—a 2× improvement in temporal stability.

Importance of initialization. Fig. 10(c, d) illustrates why initialization is critical for gradient-based pose inversion: on a simple standing pose from SAM 3D Body, 100 Adam iterations without initialization converge to a local minimum with entirely incorrect limb placement, while the analytical solver recovers the pose in a single pass with near-perfect overlay.

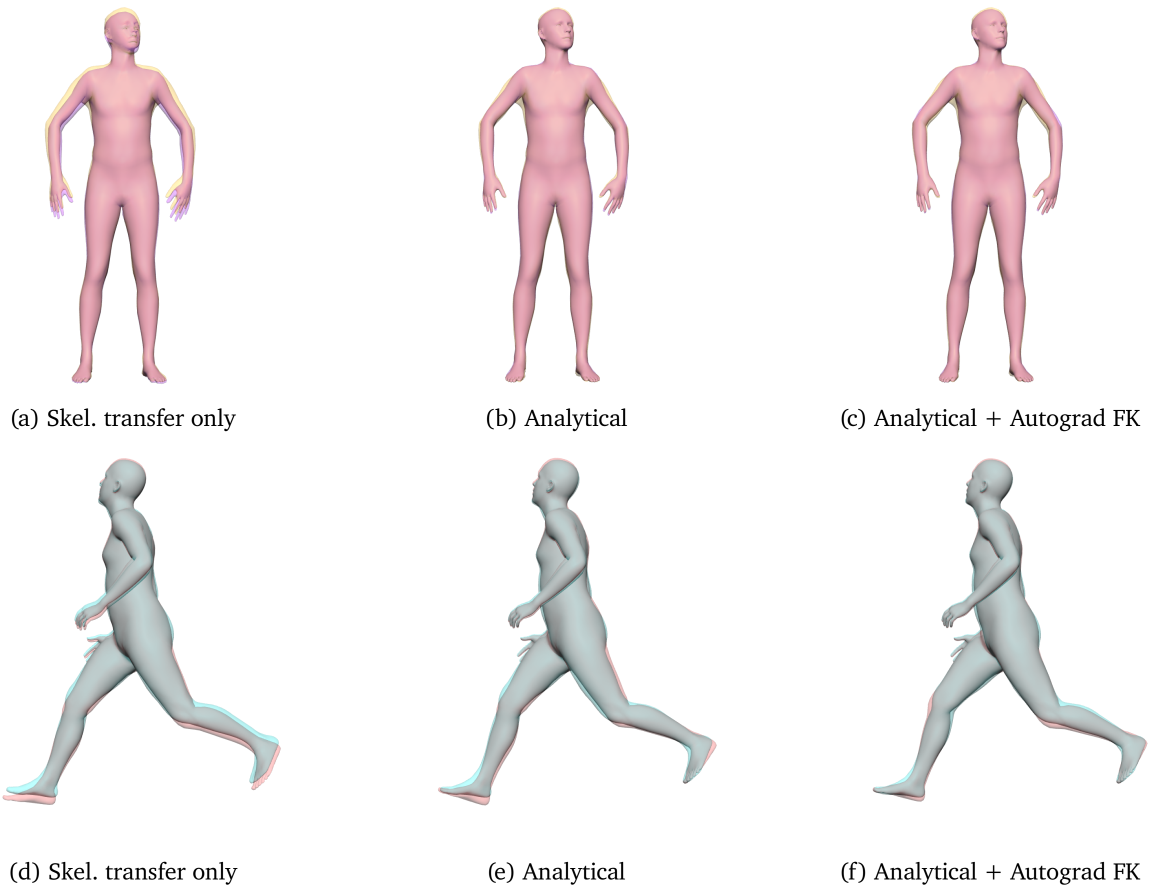
4.3. Runtime Performance

SOMA integrates directly into large-scale foundation model training loops, so the forward pass must be highly optimized. Tab. 4 reports throughput and latency across batch sizes and execution modes. Measurements were conducted on a single NVIDIA A100 80 GB GPU (Warp path) and a 32-core AMD EPYC 7763 CPU (PyTorch path), with the mid-resolution SOMA mesh and a SOMA-native identity backend (which skips the topology abstraction step).

The skeleton fitting step (RBF regression + Kabsch) takes under 1.5 ms on GPU regardless of batch size, demonstrating that the pre-factored sparse matrix approach scales efficiently. The Warp GPU path achieves over 7,000 meshes/sec at batch size 128. Adding a topology abstraction step (SMPL or MHR backend) incurs approximately 0.3–0.8 ms additional latency on GPU, which is negligible relative to total pipeline cost.

4.4. Cross-Model Shape-Space Comparison

One of the key benefits of SOMA's abstraction layers is principled cross-model comparison. We demonstrate this by evaluating four PCA shape models on 33 held-out body scans from an independent capture pipeline


(a) Skel. transfer only

(b) Analytical

(c) Analytical + Autograd FK

(d) Skel. transfer only

(e) Analytical

(f) Analytical + Autograd FK

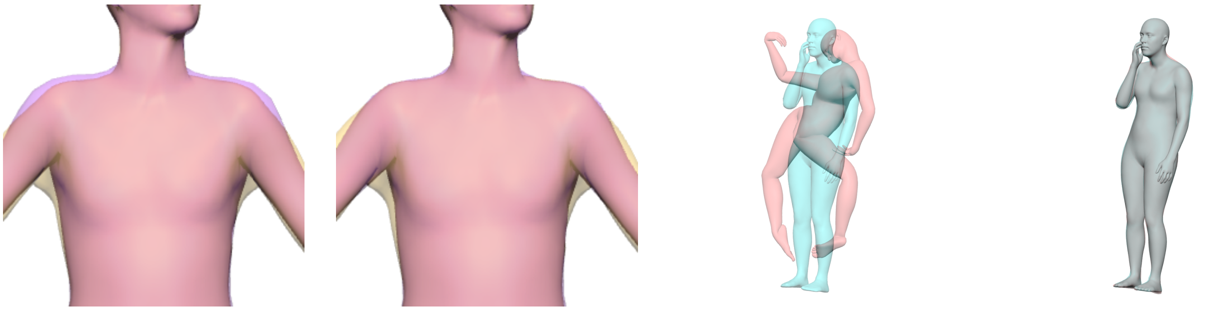
Figure 9: **Pose inversion quality across methods.** Top row: SMPL backend (yellow = ground truth, purple = SOMA reconstruction). Bottom row: MHR backend on SAM 3D Body (teal = ground truth, red = SOMA reconstruction). (a, d) Skeleton transfer alone provides a coarse estimate with visible misalignment at the extremities. (b, e) Analytical refinement with Newton-Schulz orthogonalization closely tracks the ground truth. (c, f) Adding autograd FK refinement achieves the tightest overlay, reducing residual error at the feet and hands.

without overlap with any model's PCA training data. This is typically infeasible without a unifying framework like SOMA, since each model defines its own topology, rest pose, and deformation parameters. For each model, every scan is transferred to the model's native mesh topology via barycentric interpolation (Sec. 3.3), reposed to the model's canonical rest pose via pose inversion (Sec. 3.6), and projected onto the model's PCA basis at full capacity. Tab. 5 reports per-vertex reconstruction error.

SMPL's 10-component basis captures coarse body proportions but leaves a 14 mm mean residual, consistent with its limited shape dimensionality. GarmentMeasurements (15 components) reduces this to 12 mm. SOMA-Shape achieves 5.8 mm mean with 128 components, closely matching SMPL-X's 5.5 mm at 300 components—demonstrating competitive expressiveness with fewer than half the parameters. Fig. 11 visualizes the per-vertex reconstruction error across models for representative scans: SMPL and GarmentMeasurements show widespread residual (red), while SOMA-Shape and SMPL-X achieve low error over most of the body surface (blue).

5. Conclusion

We have presented SOMA, a unified framework that decouples identity representation from pose parameterization across heterogeneous parametric body models. By mapping all supported backends to a single canonical mesh topology and rig, SOMA reduces the $O(M^2)$ per-pair adapter problem to $O(M)$ single-backend connectors, enabling practitioners to freely mix identity sources and pose data at inference time. Three abstraction layers—



(a) SVD (b) Newton-Schulz (c) Autograd (no init) (d) Analytical

Figure 10: **Pose inversion ablations.** (a, b) Shoulder zoom from a single SMPL frame (yellow = GT, purple = SOMA). SVD-based Kabsch alignment exhibits shoulder popping due to a singular-vector sign flip; Newton-Schulz produces a smooth result. (c, d) MHR on SAM 3D Body (teal = GT, red = SOMA). Without initialization, 100 autograd iterations converge to an incorrect local minimum; the analytical solver recovers the pose correctly.

Table 4: **Runtime performance.** Throughput (meshes/sec) and per-call latency (ms) of the SOMA forward pass. Breakdown shows skeleton fitting cost (RBF regression + Kabsch) vs. total forward pass. Warp = GPU path (NVIDIA Warp LBS kernel); PyTorch = CPU dense path. Identity backend: SOMA-native (no topology abstraction overhead).

Mode	**Batch**	**Skel. (ms)**	**Total (ms)**	**Meshes/sec**
Warp (GPU)	1	0.8	2.1	476
Warp (GPU)	8	0.9	3.4	2,353
Warp (GPU)	32	1.1	6.8	4,706
Warp (GPU)	128	1.4	18.2	7,033
PyTorch (CPU)	1	3.2	12.1	83
PyTorch (CPU)	8	4.1	38.7	207
PyTorch (CPU)	32	5.9	148.0	216

mesh topology, skeleton, and pose—unify heterogeneous body shapes into a single identity representation, adapt the skeleton to arbitrary identities and pose them with unified corrective deformations shared across all backends, and recover unified skeleton rotations directly from posed vertices without custom retargeting. The entire pipeline is fully differentiable, GPU-accelerated, and requires no per-model training or iterative optimization. Evaluation across multiple backends and 100 diverse identities demonstrates sub-millimeter mean topology transfer errors on body vertices, sub-centimeter pose inversion accuracy at over 300 FPS on a laptop GPU, and forward-pass throughput exceeding 7,000 meshes/sec at batch size 128.

Limitations. Several limitations remain. First, topology transfer quality depends on the quality of the source model's canonical mesh and the SOMA wrap registration: poorly registered wraps or source models with extreme vertex density asymmetry can degrade topology abstraction accuracy. Second, despite pose-dependent correctives, standard LBS still produces artifacts at highly non-rigid deformations (*e.g.* extreme elbow flexion, shoulder abduction beyond 90 degrees); learned correctives mitigate but do not eliminate these. Third, adding a new identity backend requires implementing a new identity model class and a one-time SOMA mesh registration using standard non-rigid registration tools; this is a modest but non-trivial engineering step. Fourth, SOMA's pose abstraction is not a general-purpose retargeter: it recovers pose through mesh vertex correspondence, so it is limited to models that share compatible human body geometry. It cannot abstract poses from characters with fundamentally different geometry or rigging structure (*e.g.* robots, non-humanoid characters); such cases require a specialized retargeting solution.

Table 5: **Cross-model PCA reconstruction on 33 held-out body scans.** Per-vertex L_2 distance (mm) between the unposed scan and the PCA reconstruction at each model's full component count K. All models are evaluated through the same SOMA pipeline, differing only in the target topology, rest pose, and PCA basis.

Model	K	**Mean (mm)**	**Median (mm)**	**P95 (mm)**
SMPL	10	14.11	12.31	30.49
GarmentMeasurements	15	11.81	10.67	24.18
SOMA-Shape (Ours)	128	5.82	4.81	13.60
SMPL-X	300	5.45	4.34	12.97

Acknowledgments

We thank Davis Rempe, Mathis Petrovich, and Sehwi Park for valuable feedback and help throughout the project. We thank Cyrus Hogg and Mike Sandrik for their support with data acquisition, and Dennis Lynch, Will Tellford, Jon Shepard, and Spencer Huang for helpful guidance during development.

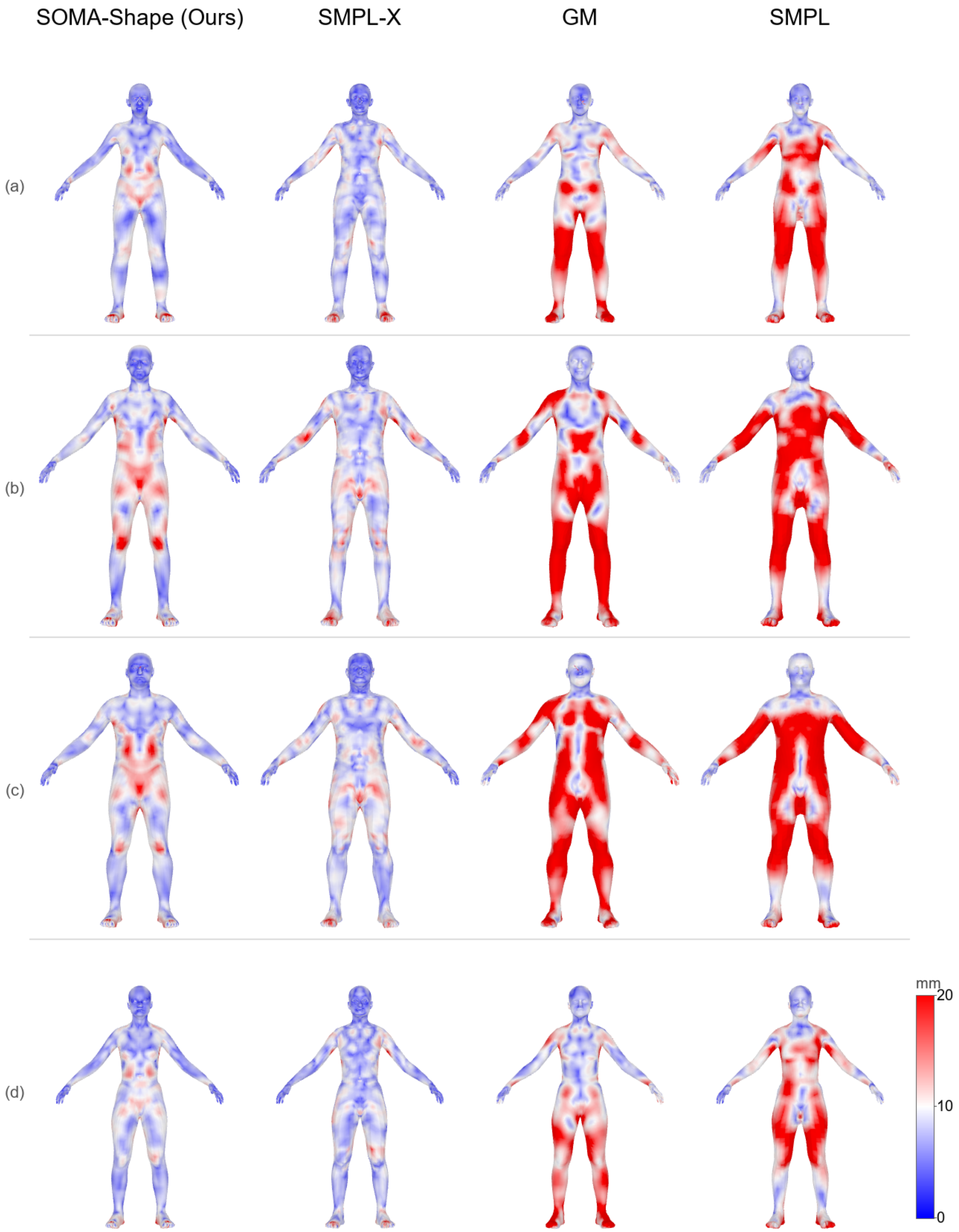


Figure 11: **Cross-model PCA reconstruction error on held-out body scans.** SOMA's topology and pose abstraction layers enable a principled cross-model comparison. Each column shows a different shape model's mesh topology, all rendered in SOMA A-pose; each row (a–d) is the *same* identity. Color encodes per-vertex L_2 error (blue = 0 mm, white = 10 mm, red $\geq$ 20 mm). SMPL (10 components) and GM (GarmentMeasurements, 15) show widespread residual, while SOMA-Shape (128) and SMPL-X (300) achieve low error across the body surface.

References

[1] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM transactions on graphics (TOG)*, volume 24, pages 408–416. ACM, 2005. 3

[2] Fabien Baradel, Matthieu Armando, Salma Galaaoui, Romain Brégier, Philippe Weinzaepfel, Grégory Rogez, and Thomas Lucas. Multi-HMR: Multi-person whole-body human mesh recovery in a single shot. *European Conference on Computer Vision*, 2024. 3

[3] Romain Brégier, Guénolé Fiche, Laura Bravo-Sánchez, Thomas Lucas, Matthieu Armando, Philippe Weinzaepfel, Grégory Rogez, and Fabien Baradel. Human mesh modeling for anny body, 2025. URL `https://arxiv.org/abs/2511.03589`. 2, 3, 5

[4] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3D human pose and shape from a video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1964–1973, 2021. 3

[5] Aaron Ferguson, Ahmed A. A. Osman, Berta Bescos, Carsten Stoll, Chris Twigg, Christoph Lassner, David Otte, Eric Vignola, Fabian Prada, Federica Bogo, Igor Santesteban, Javier Romero, Jenna Zarate, Jeongseok Lee, Jinhyung Park, Jinlong Yang, John Doublestein, Kishore Venkateshan, Kris Kitani, Ladislav Kavan, Marco Dal Farra, Matthew Hu, Matthew Cioffi, Michael Fabris, Michael Ranieri, Mohammad Modarres, Petr Kadlecek, Rawal Khirodkar, Rinat Abdrashitov, Romain Prévost, Roman Rajbhandari, Ronald Mallet, Russell Pearsall, Sandy Kao, Sanjeev Kumar, Scott Parrish, Shoou-I Yu, Shunsuke Saito, Takaaki Shiratori, Te-Li Wang, Tony Tung, Yichen Xu, Yuan Dong, Yuhua Chen, Yuanlu Xu, Yuting Ye, and Zhongshi Jiang. MHR: Momentum human rig, 2025. URL `https://arxiv.org/abs/2511.15586`. 2, 3, 5, 8

[6] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14783–14794, 2023. 3

[7] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Reconstructing and tracking humans with transformers. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 3

[8] Umar Iqbal, Kevin Xie, Yunrong Guo, Jan Kautz, and Pavlo Molchanov. KAMA: 3D keypoint aware body mesh articulation. In *3DV*, 2021. 3

[9] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976. 8

[10] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018. 3

[11] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. VIBE: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5253–5263, 2020. 3

[12] Muhammed Kocabas, Chun-Hao P Huang, Otmar Hilliges, and Michael J Black. PARE: Part attention regressor for 3D human body estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11127–11137, 2021. 3

[13] Muhammed Kocabas, Ye Yuan, Pavlo Molchanov, Yunrong Guo, Michael J. Black, Otmar Hilliges, Jan Kautz, and Umar Iqbal. PACE: Human and motion estimation from in-the-wild videos. In *3DV*, 2024. 3

[14] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2252–2261, 2019. 3

[15] Maria Korosteleva and Olga Sorkine-Hornung. GarmentCode: Programming parametric sewing patterns. *ACM Transaction on Graphics*, 42(6), 2023. doi: 10.1145/3618351. SIGGRAPH ASIA 2023 issue. 2, 5

[16] Zdislav V. Kovarik. Some iterative methods for improving orthonormality. *SIAM Journal on Numerical Analysis*, 7(3):386–389, 1970. 10

[17] Jiefeng Li, Jinkun Cao, Haotian Zhang, Davis Rempe, Jan Kautz, Umar Iqbal, and Ye Yuan. Genmo: A generalist model for human motion. *arXiv preprint arXiv:2505.01425*, 2025. 2, 3

[18] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 34(6): 248:1–248:16, October 2015. 2, 3, 5

[19] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *ICCV*, 2019. 9, 12

[20] Ahmed A A Osman, Timo Bolkart, and Michael J. Black. STAR: A sparse trained articulated human body regressor. In *European Conference on Computer Vision (ECCV)*, pages 598–613, 2020. URL `https://star.is.tue.mpg.de`. 3

[21] Priyanka Patel and Michael J. Black. Camerahmr: Aligning people with perspective. *International Conference on 3D Vision (3DV)*, 2025. 3

[22] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3, 5

[23] Mathis Petrovich, Or Litany, Umar Iqbal, Michael J. Black, Gül Varol, Xue Bin Peng, and Davis Rempe. Multi-track timeline control for text-driven 3d human motion generation. In *CVPR Workshop on Human Motion Generation*, 2024. 3

[24] Leonid Pishchulin, Stefanie Wuhrer, Thomas Helten, Christian Theobalt, and Bernt Schiele. Building statistical shape spaces for 3d human modeling. *Pattern Recognition*, 2017. 3

[25] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), November 2017. 3

[26] David A. Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental learning for robust visual tracking. *IJCV*, 77(1–3):125–141, 2008. 5

[27] István Sárándi and Gerard Pons-Moll. Neural localizer fields for continuous 3d human pose and shape estimation. *Advances in Neural Information Processing Systems*, 37:140032–140065, 2024. 3

[28] Zehong Shen, Huaijin Pi, Yan Xia, Zhi Cen, Sida Peng, Zechen Hu, Hujun Bao, Ruizhen Hu, and Xiaowei Zhou. World-grounded human motion recovery via gravity-view coordinates. In *SIGGRAPH Asia*, 2024. 2, 3

[29] Soyong Shin, Juyong Kim, Eni Halilaj, and Michael J Black. WHAM: Reconstructing world-grounded humans with accurate 3D motion. *arXiv preprint arXiv:2312.07531*, 2023. 3

[30] [TC]^2. Sizeusa: The national sizing survey. Technical report, Textile/Clothing Technology Corporation, Cary, NC, 2004. URL `http://www.sizeusa.com`. 5

[31] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. In *ICLR*, 2023. 3

[32] Triplegangers. Triplegangers 3d scans. `https://triplegangers.com`, 2025. Accessed: 2025. 5

[33] Yufu Wang, Yu Sun, Priyanka Patel, Kostas Daniilidis, Michael J Black, and Muhammed Kocabas. Prompthmr: Promptable human mesh recovery. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1148–1159, 2025. 3

[34] Yufu Wang, Evonne Ng, Soyong Shin, Rawal Khirodkar, Yuan Dong, Zhaoen Su, Jinhyung Park, Kris Kitani, Alexander Richard, Fabian Prada, and Michael Zollhofer. Duomo: Dual motion diffusion for world-space human reconstruction. *arXiv preprint arXiv:2603.03265*, 2026. 2, 3

[35] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. GHUM & GHUML: Generative 3D human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193, 2020. 3

[36] Xitong Yang, Devansh Kukreja, Don Pinkus, Anushka Sagar, Taosha Fan, Jinhyung Park, Soyong Shin, Jinkun Cao, Jiawei Liu, Nicolas Ugrinovic, Matt Feiszli, Jitendra Malik, Piotr Dollar, and Kris Kitani. Sam 3d body: Robust full-body human mesh recovery. *arXiv preprint arXiv:2602.15989*, 2026. 3, 9, 12

[37] Ye Yuan, Umar Iqbal, Pavlo Molchanov, Kris Kitani, and Jan Kautz. Glamr: Global occlusion-aware human mesh recovery with dynamic cameras. In *CVPR*, 2022. 3

[38] Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. Physdiff: Physics-guided human motion diffusion model. In *ICCV*, 2023. 3

[39] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022. 3

[40] Mingyuan Zhang, Daisheng Jin, Chenyang Gu, Fangzhou Hong, Zhongang Cai, Jingfang Huang, Chongzhi Zhang, Xinying Guo, Lei Yang, Ying He, et al. Large motion model for unified multi-modal motion generation. In *ECCV*, 2024. 2, 3

[41] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019. 8, 10