University of Thessaly

Department of Informatics and Telecommunications

Doctoral Dissertation

AI-Based Synthesis for Complex Human Poses, Motions, and Interactions

by

Georgios Albanis

Lamia, February 2026

Three-Member Supervisory Committee

Chairperson of the Supervisory Committee:

Dr. Konstantinos Kolomvatsos
Associate Professor, University of Thessaly
Department of Informatics and Telecommunications

Member of the Supervisory Committee:

Dr. Efstathios Hadjiefthymiades
Professor, National and Kapodistrian University of Athens
Department of Informatics and Telecommunications

Member of the Supervisory Committee:

Dr. Christos Anagnostopoulos
Associate Professor, University of Glasgow
School of Computing Science

Seven-Member Evaluation Committee

Dr. Konstantinos Kolomvatsos
Associate Professor, University of Thessaly
Department of Informatics and Telecommunications

Dr. Efstathios Hadjiefthymiades
Professor, National and Kapodistrian University of Athens
Department of Informatics and Telecommunications

Dr. Christos Anagnostopoulos
Associate Professor, University of Glasgow
School of Computing Science

Dr. Vassilios Plagianakos
Professor, University of Thessaly
Department of Computer Science and Biomedical Informatics

Dr. Gerasimos Potamianos
Associate Professor, University of Thessaly
Department of Electrical and Computer Engineering

Dr. Antonios Argyros
Professor, University of Crete
Computer Science Department

Dr. Nikolaos Tziritas
Associate Professor, University of Thessaly
Department of Informatics and Telecommunications

"Nothing happens until something moves"

— Albert Einstein

AI-Based Synthesis for Complex Human Poses, Motions, and Interactions

Abstract

Motion Capture (MoCap) is a technology with broad applications, yet its democratization remains an open research topic. Even high-end optical systems require laborious manual cleanup, while recent advances in lower-cost, markerless MoCap are hampered by poor data quality, making them almost useless for downstream applications. This Thesis addresses these challenges, taking a shift from existing solutions and leveraging AI synthesis, enabling the accurate capture of complex poses, motion, and interactions.

First, we tackle the challenge of complex human poses by using representation learning to synthesize new training data, balancing existing MoCap datasets to train more effective AI models. We demonstrate the efficacy of this approach on the task of automatic marker labeling, a critical step in optical MoCap workflows.

However, these models can introduce complex noise patterns, which, combined with low-cost sensors often used in real-world settings, lead to significant uncertainty in measurements. Existing optimization approaches often assume clean data or simple noise models, making them ill-suited for these scenarios. We, therefore, propose a novel optimization framework that models the uncertainty of the constraints themselves, learning it alongside the measurements.

While effective for optical MoCap, this method is insufficient for the far more challenging case of markerless data. Markerless MoCap suffers from severe artifacts, including jittery joint estimates, swapped body parts, and completely missing data, which are intractable for most solvers. Furthermore, most approaches fail to leverage the temporal coherence present in motion data. Hence, we propose a framework for robustly capturing motions, which leverages a learned manifold with specific geometric properties to represent the space of valid human poses. This enables us to introduce novel synthesis techniques that inherently leverage temporal coherence to enable efficient motion solving while effectively alleviating severe artifacts.

In summary, this Thesis presents a suite of innovative techniques that utilize representation learning to synthesize new samples for balancing training data, model complex noise patterns, and robustly solve challenging motion capture scenarios, thereby taking a step towards making high-quality MoCap widely accessible.

Contents

List of Figures

List of Tables

Frequently Used Symbols

$\mathcal{B}(\boldsymbol{\beta}, \boldsymbol{\theta})$	SMPL body model function
$\bar{T}$	template mesh
V	number of vertices in the mesh
$\mathbf{v}_k(\boldsymbol{\theta})$	posed position of the k-th mesh vertex
N_j	number of bones in the kinematic tree
$\boldsymbol{\beta}$	body shape blend-shape coefficients
$\boldsymbol{\theta}$	pose parameters
$\mathbf{T}$	global root transformation
$\mathbf{W}$	linear blend skinning weights
$\mathbf{S}_P(\boldsymbol{\theta})$	pose-dependent corrective blend-shape matrix
$\mathcal{J}(\boldsymbol{\beta}, \boldsymbol{\theta})$	landmark regressor function
J	number of joints/landmarks
$\mathcal{M}$	learned latent manifold
$\mathbf{z}$	latent vector
$\mathcal{E}$	encoder
$\mathcal{G}$	generator / decoder
A	autoencoder
S^n	n-dimensional unit hypersphere
$T_{\mathbf{z}}\mathcal{M}$	tangent space of manifold $\mathcal{M}$
$\mathbf{g}_{\text{tangent}}$	projected Riemannian gradient on $\mathcal{M}$
$\mathcal{E}_{\text{data}}$	data fitting objective
$\mathcal{E}_{\text{prior}}$	prior / regularization objective
$\boldsymbol{\pi}(\cdot)$	camera projection function
$\boldsymbol{\ell}_{\text{est}}$	estimated 3D landmarks
$\boldsymbol{\ell}_i^*$	i-th model-predicted landmark
σ_i	per-landmark learned uncertainty
$\mathcal{T}$	temporal window
$\rho(\epsilon)$	relevance / sample weighting function
$\mathcal{S}(\cdot)$	sampling operator in latent space
$\mathcal{A}$	anchor poses used for sampling

Acknowledgments

I wish to express my profound gratitude to my supervisor, Professor Kostas Kolomvatsos. His invaluable guidance, unwavering patience, and steadfast motivation were instrumental throughout every stage of this research. He fostered an environment of intellectual curiosity that was essential for the writing of this Thesis, and I could not have asked for a better advisor and mentor for my Ph.D. study.

I am also profoundly grateful to my Moverse co-founders, Argyris, Nick, and Spiros. Embarking on a startup journey concurrently with my doctoral studies presented unique challenges, and I owe them a debt of gratitude for their remarkable support, understanding, and for affording me the space and time necessary to complete this work. A special note of appreciation is reserved for Nick. He was an invaluable intellectual partner, always available to discuss ideas, provide critical feedback, and offer assistance in any way possible. I am thankful for his constant encouragement, and even for our disagreements, which compelled me to think more critically and deeply about my research.

I would also like to thank my teachers throughout the years, whose guidance and encouragement shaped me into the person I am today and inspired me to pursue the path that led me here. Without their support, I would not be who I am today.

Finally, this accomplishment would have been impossible without the foundational support of my family and friends. I extend my heartfelt thanks to my parents, Nikos and Anna, and my brothers, Thomas, Dimitris, and Raphael, for their unlimited support and belief in me.

Above all, I thank Stavroula. Her enduring patience and love were my constant anchor, sustaining me through the most challenging times of this journey.

To my parents, brothers, and Stavroula.

Chapter 1

Introduction

1.1 Motivation

Motion Capture (MoCap) —the process of digitizing human motion —has wide applicability, ranging from virtual characters in films and video games to precision analysis in sports and rehabilitation as depicted in Figure 1.1. The roots of MoCap trace back to Etienne Jules Marey's chronophotography experiments in the late nineteenth century and Muybridge's pioneering motion studies, as shown in Figure 1.2, evolving through analog mechanical systems in the mid-20th century to today's sophisticated digital pipelines. Modern optical marker-based systems (e.g., Vicon, Optitrack, Qualisys, PhaseSpace) are considered the most accurate solutions but require controlled environments, expensive hardware, and experienced personnel to operate them. Apart from that, in cases with complex poses where the tracking accuracy of the markers is compromised, a time-consuming manual post-processing step is required. Inertial measurement devices (IMUs) embedded in wearable suits (e.g., Rokoko, and Xsens) free subjects from studio constraints, as they do not require cameras or markers, but they suffer from accumulated drift and magnetic interference. Recently, markerless approaches leveraging multi-view RGB/RGB-D sensors and deep learning (e.g., Captury, Move AI , Moverse, Theia Markerless) promise plug-and-play convenience yet remain sensitive to occlusions, clothing variations, and lighting conditions.

Despite decades of progress, a gap still separates laboratory-grade accuracy from consumer-grade accessibility. Closing this gap would democratize MoCap— e.g., allowing athletes to analyze their technique on a smartphone, animators to iterate in real time on set, and clinicians to conduct remote assessments without requiring specialized labs. This Thesis contributes to that goal by developing novel methods

Figure 1.1 MoCap applications ranging from movies, and location-based entertainment (i.e., LBEs) to sports science and rehabilitation.

for robust motion capture under challenging conditions, complex poses and interactions, including low-cost sensors and markerless setups. We leverage representation learning to map human poses into compact latent spaces and explicitly model uncertainty during optimization, enabling reliable performance outside traditional lab environments.

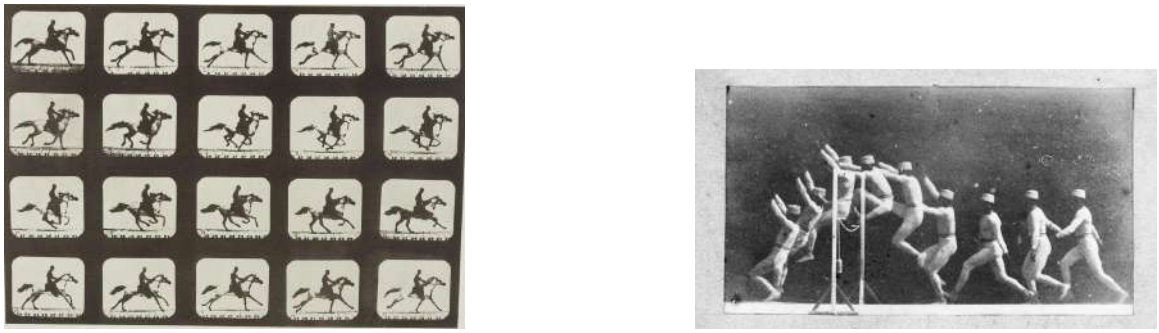


Figure 1.2 Motion Capture origins. On the left, photographs by Eadweard Muybridge depicting a horse in motion, and on the right, a photograph showing the phases of movement of a man jumping a hurdle, made by Étienne-Jules Marey (1892).

1.2 Challenges

Optical MoCap systems remain the gold standard in the field of digitizing motion in terms of MoCap quality (see Table 1.1). However, their utility is often constrained by practical and technical limitations. The high capital expenditure required for multi-camera arrays, specialized suits, and controlled studio environments makes these systems inaccessible to the broad community [16]. Beyond the initial cost, the data acquisition and processing pipeline presents significant bottlenecks. The process is highly susceptible to marker occlusion, where one or more markers are temporarily hidden from the cameras' view due to the subject's own body or other objects. This requires laborious and time-consuming manual post-processing, where technicians must painstakingly interpolate or reconstruct the missing marker trajectories. For instance, as shown in [16], the manual cleanup for a sequence of 24 minutes took approximately 47 hours, even from an experienced technician, highlighting the inefficiency of current pipelines.

Feature	**High-end optical**	**Low-end optical**	**Markerless**
Cost	●●●●●	●●●○○	●●○○○
Setup effort	●●●●●	●●●○○	●○○○○
Space needs	●●●●●	●●●○○	●○○○○
Cleanup effort	●●○○○	●●●●○	●●●●●
Data fidelity	●●●●●	●●●○○	●●○○○

Table 1.1: **Motion capture systems comparison.** More filled circles indicate higher magnitude along each dimension: higher cost, greater setup effort and space requirements, more cleanup effort, or higher data fidelity. High-end optical systems provide the greatest accuracy but require specialized, space-intensive installations and experienced personnel to operate. Low-end optical systems reduce infrastructure demands but suffer from occlusion and moderate cleanup. Markerless systems require minimal setup and can operate anywhere, though they typically produce noisier outputs, such as jitter and sliding, requiring extensive cleanup to be used in downstream applications.

This manual cleanup is not only a drain on resources but also introduces a potential for subjective error. The integration of Artificial Intelligence (AI) offers a promising solution to these issues, with the potential to reduce the required number of sensors or even automate the marker cleanup process. However, this approach is

Figure 1.3 Motion data suffers from redundancy as repetitive motions are common, while many parts of the motion cycle repeat. For instance, in this side kick motion, the start and end frames contain similar (i.e., idle poses).

not without limitations, as the performance of AI models is fundamentally dependent on the quality and characteristics of the training data. This dependency reveals two critical challenges:

- **The Long-Tailed Data Distribution**: Existing MoCap datasets are heavily imbalanced, as a motion cycle, even across different types of motion, contains mostly repetitive poses, as shown in Figure 1.3. They suffer from a long-tailed distribution where common, simple poses (e.g., walking, standing still) are over-represented, while complex, dynamic, or rare poses are scarce (Figure 1.4). This data redundancy biases AI models, leading them to perform exceptionally well on common motions, but fail significantly when confronted with the infrequent yet often more interesting or critical actions.
- **High-Dimensional Data Balancing**: Conventional methods for balancing datasets from other domains (e.g., image classification) are not effective for high-dimensional human pose data. Simple augmentation techniques like over-sampling rare poses are not straightforward. At the same time, synthetic data generation methods can easily create physically impossible or kinematically flawed movements, corrupting the training data with unrealistic examples.

Apart from that, reducing the capital expenditure on a MoCap system by lowering the number of cameras and their quality introduces a higher level of measurement noise and greater uncertainty in the solving process, as depicted in Figure 1.5. When AI models are used to fill in the gaps from this sparser data, they introduce their own layer of model-induced noise or prediction error. This creates a compounded error effect where both hardware limitations and algorithmic imperfections degrade the final output quality. Modeling this complex, multi-source noise is a significant challenge. Robust statistical methods [17, 18], such as Gaussian Processes and other

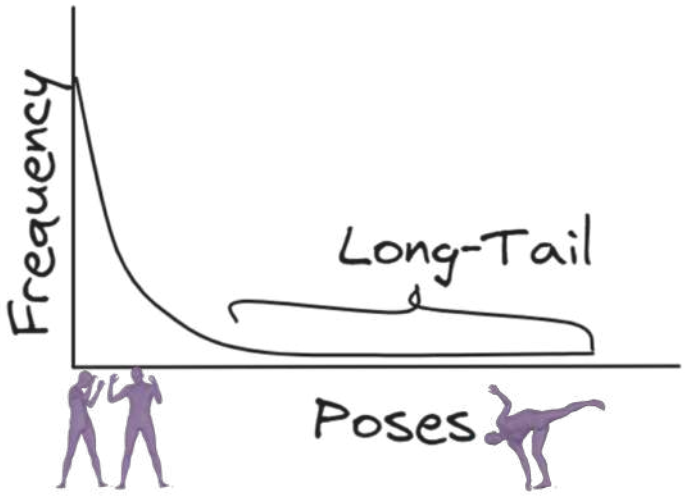


Figure 1.4 The long-tailed distribution issue, where rare poses are less frequent than common ones, such as standing still.

kernel-based approaches [19], have been widely used for modeling uncertainty in time-series data. However, they often assume that the noise follows a known, well-behaved distribution (e.g., Gaussian). In reality, the noise profile of a sparse AI-driven MoCap system is often unknown beforehand, non-stationary, and non-Gaussian, limiting the effectiveness of these conventional modeling techniques.

On the other hand, markerless MoCap systems represent a paradigm shift towards greater accessibility and ease of use. By relying on computer vision algorithms to track body keypoints from video, they eliminate the need for specialized suits and markers. However, this flexibility comes at the cost of robustness and data fidelity, especially when compared to their marker-based counterparts. These systems often struggle under real-world conditions, with their performance degrading due to:

- Varying or challenging lighting conditions.
- Occlusions, including both self-occlusion and occlusions from external objects.
- A wide diversity of body shapes, sizes, and clothing.

A primary issue plaguing markerless systems is the lack of temporal consistency, which manifests as several distinct and problematic artifacts:

- **High-Frequency Jitter**: Predicted keypoints often exhibit rapid, unnatural oscillations around their true position, resulting in shaky and unusable motion data.

Figure 1.5 Different forms of noise are generated by the sensors (on the left side of the image) or by the AI model (in the middle). Low-cost sensors introduce measurement noise and ghosting artifacts, whereas sparse camera views cause information loss. When AI models treat observations as fully certain, this mismatch propagates uncertainty, leading to unstable or uncertain predictions. Existing solving methods, including Mosh [1], and Mosh++ [2]. On the right, ground truth and results are heavily affected by the higher level of noise, leading to inaccurate results.

- **Missing Keypoints**: Keypoints may disappear entirely for several frames when a body part is occluded, resulting in data gaps.
- **Inconsistent Predictions**: The position of a tracked joint can suddenly jump or drift between adjacent frames, breaking the illusion of smooth, continuous motion.
- **Limb Swapping/Flipping**: Symmetric body parts, such as the left and right legs or arms, can have their identities "flipped," a critical error that is difficult to detect and correct automatically.

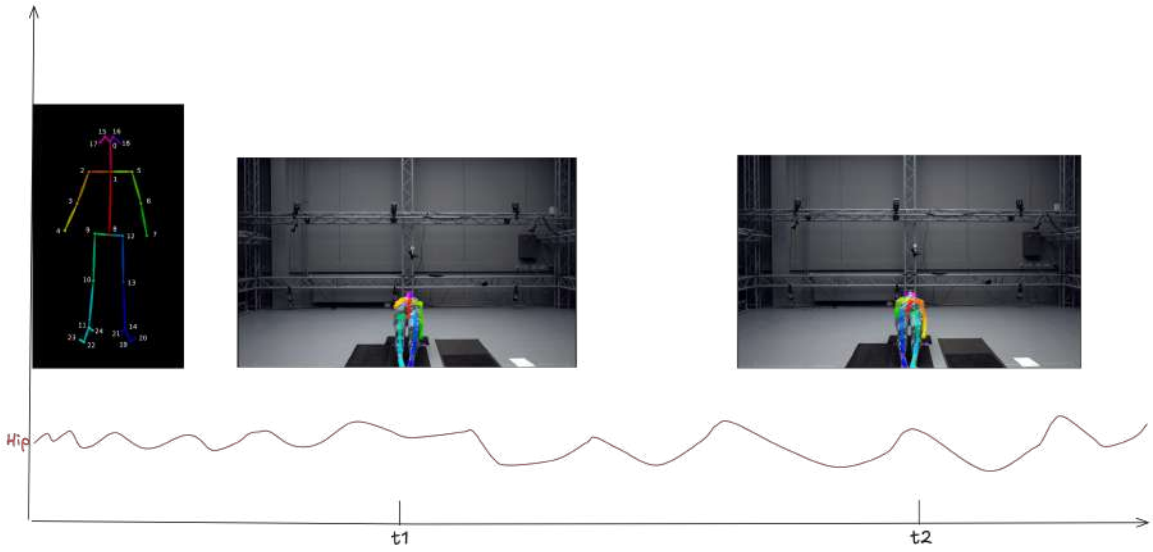


Figure 1.6 Challenges in markerless motion capture. The color-coded OpenPose skeleton (*left*) illustrates the tracked joints used for visualization. Consecutive frames (t_1, t_2) show a yoga-like motion where distinct failure cases emerge. The hip trajectory (red curve) exhibits **high-frequency jitter**, causing unnatural oscillations over time, while specific joints become **missing** due to intermittent occlusion. The alternating colors of the legs (blue/cyan) highlight **limb swapping or flipping**, where symmetric limbs are misidentified between frames. Together, these artifacts represent a key challenge in markerless MoCap systems.

These errors, see Figure 1.6, are particularly exaggerated in sparse multi-view setups. When using only a few low-cost cameras, the system has less information available to disambiguate challenging poses, resolve occlusions, or accurately triangulate 3D joint positions. Consequently, the final 3D reconstruction is often a compound of multiple 2D prediction errors, leading to a final output that is significantly less reliable than one produced in a controlled, multi-camera environment.

1.3 Contributions

Aiming to mitigate such challenges, we make the following core contributions to the field of human digitisation:

- **Firstly**, we propose techniques for balancing MoCap datasets to better represent rare and complex poses. Unlike existing methods that rely on reconstruction error metrics—which often fail to distinguish rare poses from common ones—and standard linear interpolation that limits diversity, we leverage representation learning for effective sampling through AI synthesis. Specifically, we utilize the Mahalanobis distance in the latent space to accurately identify rare samples automatically and employ high-order interpolation schemes (e.g., SQUAD) to synthesize diverse and physically plausible data. This allows us to train effective AI models, reducing the need for manual marker cleanup, even in challenging motion sequences.

- **Secondly**, we develop an uncertainty-aware solver that improves robustness in noisy marker-based motion capture settings. In contrast to traditional solvers that employ fixed robust loss functions or assume Gaussian noise distributions—which struggle with the compounded, non-stationary noise of low-cost sensors and AI inference uncertainty—we represent uncertainty as a learnable parameter for each landmark. By optimizing this uncertainty jointly with the data and prior terms, our system self-calibrates to automatically down-weight unreliable inputs. We validate this approach on both synthetic benchmarks and real-world captures, demonstrating significant improvements in reconstruction accuracy and robustness.

- **Thirdly**, we introduce a simple yet effective framework for temporally consistent motion solving. Diverging from state-of-the-art approaches that depend on complex multi-stage optimization and explicit temporal smoothness terms to mitigate jitter, we propose a single-stage bundle solver. We leverage a novel hyperspherical pose prior (SPoser) and SIDER2 interpolation to synthesize motion through latent keyframe optimization. This enforces implicit temporal smoothness without explicit regularization, enabling high-fidelity human motion reconstruction with fewer computational resources.

1.4 Publications

The methods introduced in this Thesis have been validated through multiple peer-reviewed publications, mentioned below in chronological order:

- **Albanis, Georgios**, Anargyros Chatzitofis, Spyridon Thermos, Nikolaos Zioulis, and Kostas Kolomvatsos. "Towards Scalable and Real-time Markerless Motion Capture." In 2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), pp. 724-725. IEEE, 2022. [20]

- **Albanis, Georgios**, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis, and Kostas Kolomvatsos. "Noise-in, Bias-out: Balanced and Real-time MoCap Solving." In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4237-4247. 2023. [7]

- **Albanis, Georgios**, Nikolaos Zioulis, and Kostas Kolomvatsos. "BundleMoCap: Efficient, Robust and Smooth Motion Capture from Sparse Multiview Videos." In Proceedings of the 20th ACM SIGGRAPH European Conference on Visual Media Production, pp. 1-9, 2023. [10]

- **Albanis, Georgios**, Nikolaos Zioulis, and Kostas Kolomvatsos. "BundleMoCap++: Efficient, Robust and Smooth Motion Capture from Sparse Multiview Videos." Computer Vision and Image Understanding. Volume 249, (2024): 104190, [21]

- **Albanis, Georgios**, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis, and Kostas Kolomvatsos. "From bias to balance: Leverage representation learning for bias-free MoCap solving." Computer Vision and Image Understanding Volume 251 (2025): 104241. [22]

- **Albanis, Georgios**, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis, and Kostas Kolomvatsos. "Robust and Efficient AI Motion Capture". In Laval Virtual Doctoral Consortium Proceedings. 2025 [23]

Additionally, showing the real-world application of this Thesis, we demonstrated two relevant demos in the following venues:

- **Albanis, Georgios**, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis,and Kostas Kolomvatsos. "MoCatalyst: Accelerating and Automating MoCap". Accepted and demonstrated as a demo in ICCV 2023. [24]

- **Albanis, Georgios**, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis, Vladimiros Sterzentsenko, and Kostas Kolomvatsos."LightMoCap: Lightweight, Real-time and Scalable Markerless Motion Capture". Demonstrated in CVMP 2023. [25]

1.5 Thesis Organization

The Thesis is structured as follows:

- **Chapter 1** contains a rough overview of the problem to target, existing solutions, advantages, and disadvantages of these methods, and the contributions made in this Thesis
- **Chapter 2** reviews human body representations, sensor modalities, and generative priors (VAEs, normalizing flows) crucial for subsequent chapters.
- **Chapter 3** surveys related work in data balancing, body solvers, temporal priors, and bundle optimization.
- **Chapter 4** details the VAE-based balancing scheme, synthetic tail generation, and extensive quantitative evaluation on long-tail datasets.
- **Chapter 5** describes architecture design, uncertainty modeling, and evaluation on both synthetic benchmarks and in-the-wild captures.
- **Chapter 6** introduces single-stage keyframe optimization, discusses algorithmic trade-offs, and presents a live system performance results.
- **Chapter 7** summarizes contributions and outlines directions toward fully markerless, consumer-grade real-time MoCap.

Chapter 2

Foundations

In this chapter, we review the major supporting technologies that form the foundation of this Thesis. We begin by introducing the parametric representation of the human body used throughout our work. Subsequently, we detail the constraints that drive the motion capture and pose estimation optimization processes. We then discuss the critical role of human pose priors not only as regularisers but also as an intermediate to improve the quality and efficiency, exploiting their inherent properties. Finally, we present the mathematical formulations for generating and manipulating motion through interpolation in learned latent spaces.

2.1 Human Modeling

The selection of a suitable representation of the human body is critical and highly application-dependent, bridging the domains of computer graphics, computer vision, and biomechanics [26]. The human form is an object of immense geometric intricacy and dynamic subtlety, capable of a wide range of shapes, poses, and motions [27]. Capturing this complexity in a computationally tractable format has led to a remarkable evolution in modeling techniques. However, there is a tradeoff between abstraction, fidelity, and computational cost. For instance, early models prioritized abstraction [28] to capture the essence of motion at a low computational price. As processing power increased, the focus shifted towards achieving higher geometric and appearance fidelity [4, 26, 27, 29]. The major paradigms that have emerged are illustrated in Figure 2.1.

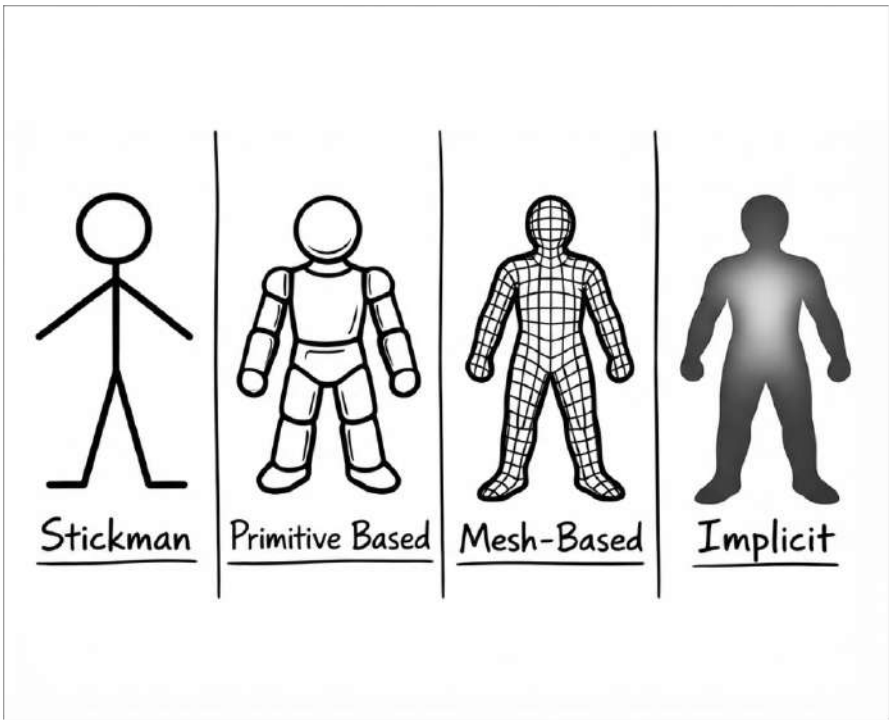


Figure 2.1 Different representations of the human body: Stick Figure, Primitive-Based, Mesh-Based, and Implicit.

2.1.1 Stick Figure Representation

The simplest representation of the human body is the stick figure model [30], where anatomical landmarks (joints) are connected using lines, representing body limbs. It's a basic abstraction that captures the essence of human movement by focusing on the movement of bones and their connections; however, it lacks the detailed surface representation necessary for realistic rendering or interaction modeling, while it cannot capture shape and appearance [28]. Thus, this body representation does not fully capture the complexities of 3D motion.

2.1.2 Primitive-Based Representation

Another simplified yet more expressive method involves using geometric primitives, such as cylinders or ellipsoids, to model body segments. This method is essentially a form of Constructive Solid Geometry (CSG), where a complex object is built by combining simpler ones. The resulting model, while not anatomically precise, effectively conveys the volume and general shape of the human figure. For instance, Marr

and Nishihara [31] introduced the use of cylindrical primitives for visual representation. Such methods provide a balance between computational efficiency and improved anatomical resemblance. However, their limited granularity prevents the simulation of detailed muscle deformation and realistic interactions, thereby constraining their utility in animation and graphics applications.

2.1.3 Mesh-Based Representation

For applications that demand high fidelity, mesh-based representations are widely employed. These models represent the body surface with interconnected triangles, enabling detailed anatomical and graphical realism [32]. However, a static mesh is merely a digital sculpture (i.e., static), which needs to be moved accordingly (i.e., to allow the static surface to bend, stretch, and deform in a seemingly organic manner). To do so, a base mesh is rigged to a kinematic tree, to enable deformation, as shown in Figure 2.2. Skinning involves associating mesh vertices with a skeleton, usually through Linear Blend Skinning (LBS) as defined in Eq. (2.1). In LBS, each vertex of the mesh is rigged to the different bones with specific weights, with a process called ***weight painting***: In this process, each vertex in the mesh is assigned a set of "blend weights" that specify the degree of influence each bone in the skeleton has on that vertex. The weights for a single vertex must sum to unity. For example, a vertex located on the bicep would be heavily influenced by the upper arm bone, slightly by the forearm bone (to allow for smooth blending across the elbow), and not at all by any bones in the legs. This process, known as weight painting, is a crucial and often time-consuming artistic task. The position of a posed vertex is, then, computed as a weighted sum of the bone transformations (i.e., blending: the final position of each vertex in the posed mesh is calculated as a weighted average of its positions as transformed by each of the influential bones). Figure 2.2 indicates a wireframe mesh, with the underlying kinematic tree and the skinning weight for the head joint. Given a mesh rigged to a kinematic tree made of N_j bones, the LBS equation is defined as follows and gives us the location of the posed vertices:

$$\mathbf{v}_k(\boldsymbol{\theta}) \;=\; \sum_{i=1}^{N_j} w_{k,i} \left(\mathbf{R}_i(\boldsymbol{\theta})\, \bar{\mathbf{v}}_k + \mathbf{t}_i(\boldsymbol{\theta}) \right), \tag{2.1}$$

where $\mathbf{v}_k(\boldsymbol{\theta}) \in \mathbb{R}^3$ is the posed position of the k-th vertex, $\bar{\mathbf{v}}_k \in \mathbb{R}^3$ is the k-th vertex of the template mesh in its rest pose, $\mathbf{R}_i(\boldsymbol{\theta}) \in \mathbb{R}^{3\times3}$ and $\mathbf{t}_i(\boldsymbol{\theta}) \in \mathbb{R}^3$ are the rotation matrix and translation vector of the i-th bone derived from the pose parameters $\boldsymbol{\theta}$, and $w_{k,i} \in \mathbb{R}$ is the skinning weight indicating the influence of the i-th bone on the k-th vertex [32].

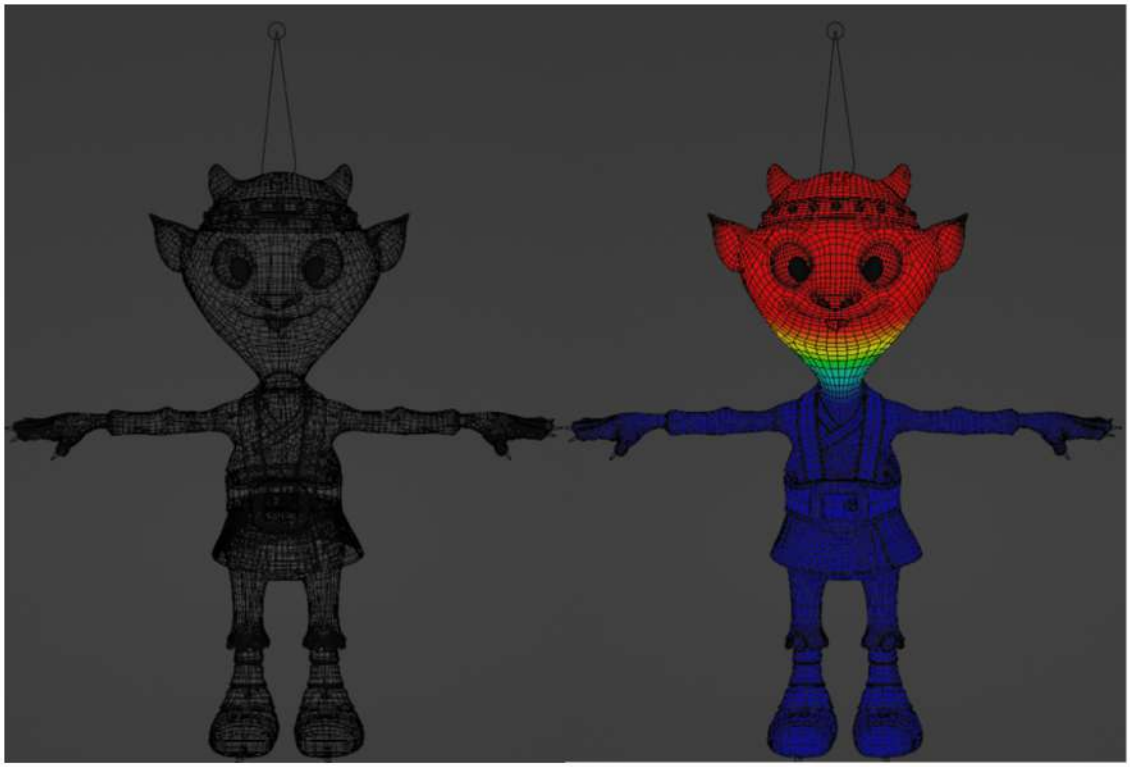


Figure 2.2 Illustration of mesh rigging. Left: The base mesh rigged to a skeleton is shown in wireframe. Right: The mesh vertices are visualized using color-coded skinning weights corresponding to the head bone, indicating the degree of influence each vertex receives during deformation (red = full influence, blue = no influence).

Although computationally efficient, LBS has limitations, notably its inability to accurately model muscle bulging during joint articulations, which can lead to artifacts in the final mesh. The most common among them is the "candy wrapper" effect, where twisting a limb (like the elbow) causes the mesh to collapse unnaturally around the axis of rotation, as depicted in Figure 2.3.

To address this, corrective blend shapes have been introduced [33]. The skinning equation thus becomes:

$$\mathbf{v}_k(\boldsymbol{\theta}) \;=\; \sum_{i=1}^{N_j} w_{k,i} \left(\mathbf{R}_i(\boldsymbol{\theta}) \left(\bar{\mathbf{v}}_k + \mathbf{s}_{p,k}(\boldsymbol{\theta}) \right) + \mathbf{t}_i(\boldsymbol{\theta}) \right), \tag{2.2}$$

where $\mathbf{s}_{p,k}(\boldsymbol{\theta}) \in \mathbb{R}^3$ denotes the k-th row of the pose-dependent corrective matrix $\mathbf{S}_P(\boldsymbol{\theta}) \in \mathbb{R}^{6890\times 3}$, representing a pose-dependent corrective offset for the k-th vertex in the rest pose, either manually crafted or learned from data. This formulation captures the essential deformation pipeline for rigged meshes, which serves as the

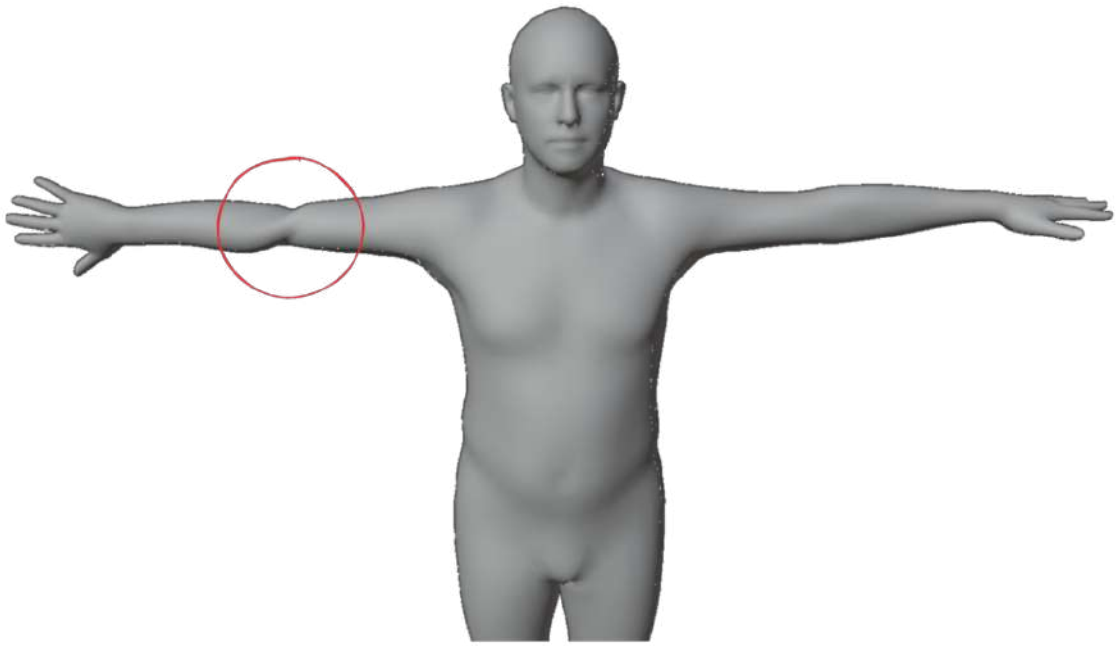


Figure 2.3 Candy-wrapper artifact, caused by LBS. Essentially, this is a loss of volume over the mesh caused mainly by the steep change of position between vertices.

basis for the statistical models discussed next.

However, in most cases, animators model the kinematic joints without respecting the human anatomy, leading to an overparameterization of the body motion, limiting their applications to biomechanics scenarios. Apart from that, personalising such models to represent a specific subject is also challenging, as changing the body shape requires editing the mesh vertices, adjusting the kinematic tree, and potentially adjusting the pose-dependent blend shapes to the new subject. Many works have attempted to automate the skeleton transfer between body shapes; however, this remains an open problem [34, 35].

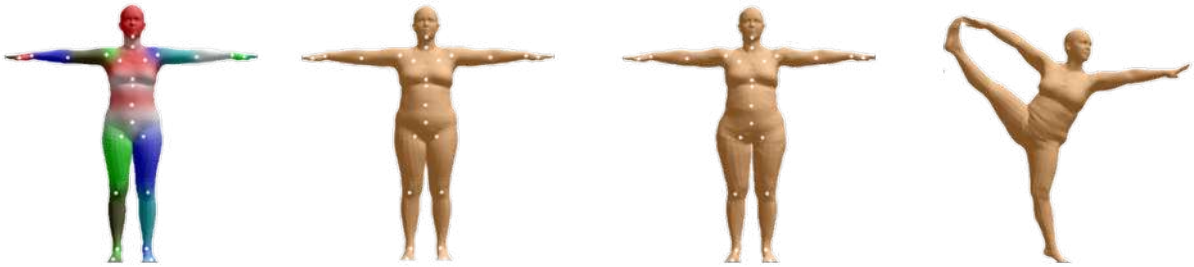
2.1.4 Statistical Mesh-Based Representation

A significant paradigm shift in human body modeling is moving from manually crafted or generic representations to models learned directly from large-scale datasets of real human scans. This data-driven approach allows for the creation of models that not only possess high-fidelity geometry but also deform in ways that are statistically consistent with the true variations in human shape and pose. Statistical Shape Models (SSMs) provide a compact geometric description of classes of objects that share se-

mantic similarities [36]. These models typically utilize Principal Component Analysis (PCA) [37] to capture the primary shape variations within a dataset. Instead of explicitly storing vertex positions for each mesh, they compute a mean shape along with principal modes of variation. This strategy enables an efficient representation of shape variability with a minimal number of parameters. SSMs have found successful applications in various domains, such as facial modeling [37], bone modeling [38], organ modeling [36], and full human body modeling.

Over the years, multiple statistical human body models have been developed, beginning with SCAPE [29], which models shape variations at the triangle level. SMPL [27] further refined this by modeling deformations at the vertex level and was subsequently expanded into SMPL-X [4], incorporating facial and hand articulation. More recent approaches include GHUM [39], which employs variational auto-encoders for comprehensive full-body modeling, STAR [40], which enhances shape expressiveness and pose-dependent deformations, and SUPR [41], specifically addressing accurate foot compression. A recent model, namely MHR [42], decouples skeleton and shape, offering a modern rig that is well-suited for AR/VR applications. These statistical models are typically created by registering a template mesh to a large set of human body scans, allowing the direct learning of soft-tissue deformations across various poses. Thus, corrective blend shapes derived from data can effectively replace manually crafted adjustments. For a thorough review of SSMs in human anatomy modeling, the reader is referred to [36].

In this Thesis, we predominantly utilize the SMPL model for human body representation. Introduced by Loper et al. [27], SMPL begins with a neutral body template, $\bar{T}$, composed of 6890 vertices, and corresponding skinning weights $\mathbf{W}$. Shape variations are controlled by a shape parameter vector $\boldsymbol{\beta} \in \mathbb{R}^{300}$, with typically only the first 10 principal components necessary to represent significant variations, thus $\boldsymbol{\beta} \in \mathbb{R}^{10}$. A per-vertex shape-dependent displacement $\mathbf{S}_S(\boldsymbol{\beta}) \in \mathbb{R}^{6890\times3}$ is calculated and added to the template body mesh. A learned regressor defines a kinematic structure comprising 24 joints, and fixed skinning weights, which rigs each vertex to this skeleton. Additionally, pose-dependent corrective blend shapes, $\mathbf{S}_P(\boldsymbol{\theta}) \in \mathbb{R}^{6890\times3}$, are applied to model realistic tissue deformations. Each pose is succinctly described using 3 rotation angles for each of the 24 joints, $\boldsymbol{\theta} \in \mathbb{R}^{72}$. The high-level pipeline of animating SMPL under a known pose is depicted in Figure 2.4, denoted as $\mathcal{B}(\boldsymbol{\beta}, \boldsymbol{\theta})$.



(a) $\bar{T}, \mathbf{W}$ (b) $\bar{T} + \mathbf{S}_S(\boldsymbol{\beta})$ (c) $\bar{T} + \mathbf{S}_S(\boldsymbol{\beta}) + \mathbf{S}_P(\boldsymbol{\theta})$ (d) $\mathcal{B}(\boldsymbol{\beta}, \boldsymbol{\theta})$

Figure 2.4 SMPL model. (a) Template mesh with color-coded blend weights and joints shown in white. (b) Mesh with identity-dependent blendshape contributions only, where vertex and joint locations vary linearly with the shape vector $\boldsymbol{\beta}$. (c) Addition of pose blendshapes illustrating deformation for a split pose, notably expanding the hips. (d) Final reposed mesh obtained via linear blend skinning for the split pose.

2.1.5 Implicit Mesh Representation

In contrast, implicit neural shape representations cast surfaces as zero-level sets of continuous 3D fields, avoiding the need to deform an initial mesh and thus inherently supporting large topology changes; explicit geometry is recovered via surface extraction algorithms such as Marching Cubes [43] or learned/differentiable variants [44, 45]. Earlier implicit formulations based on classical level sets also provided flexible topology handling [46]. While voxelized versions of these implicit fields incur prohibitive memory costs and limit resolution, recent work replaces them with neural implicit functions — for example, occupancy networks [47] or learned signed distance functions such as DeepSDF [48], differentiable rendering of SDFs [49], and enhanced coordinate encodings [50] — that compress shape information into optimizable latent codes. Related strategies that blend explicit and implicit geometry to retain topological flexibility while improving efficiency have also been explored, e.g., Shape-As-Points [51]. These representations have been applied successfully to the scene-level object modeling like furniture [48, 52] and to learning human shape/pose priors [53, 54]. However, their potential for applications where fine-scale geometric details govern physical performance, such as sheet stamping of 3D components, has not been sufficiently investigated, leaving a gap in the topology-aware modeling of small functional features. More recently, human-centric 3D Gaussian Splatting representations have emerged as an efficient alternative to dense volumetric implicit fields for dynamic human modeling. Early extensions of Gaussian Splatting to articulated humans demonstrate that splatted anisotropic primitives can serve not only as rendering

accelerators but also as structured geometric representations. Gaussian Avatars [55] enable real-time animatable head reconstruction by conditioning Gaussian primitives on pose and expression. Human Gaussian Splatting [56] extends this paradigm to full-body dynamic reconstruction by anchoring Gaussians in a canonical space and deforming them via articulated priors. More general dynamic Gaussian frameworks, such as 4D Gaussian Splatting [57] and dynamic Gaussian deformation models incorporate explicit spatio-temporal deformation mechanisms that are widely adopted in human-centric extensions. Building upon these foundations, recent works focus on improving generalization under sparse supervision, including SinGS [58], RoGSplat [59], NSGHG [60], and HuGDiffusion [61], which introduce kinematic priors, robust multi-view generalization, neural surface guidance, and diffusion-based Gaussian parameter generation, respectively. Collectively, these approaches replace dense implicit field evaluations with differentiable Gaussian primitives optimized jointly with pose and shape, enabling faster training and real-time rendering while preserving geometric fidelity under sparse or noisy observations.

2.2 Representation of Human Body

Indeed, SMPL allows us to effectively model human movement as a function of only two parameters $\mathcal{B}(\boldsymbol{\beta}, \boldsymbol{\theta})$; its range of values (i.e., the plausible human poses) lies in an unstructured, high-dimensional, and infinite data space. It is of paramount importance to find a proper mapping of this high-dimensional space to a lower-dimensional structure space. Essentially, what we try to do is depicted in Figure 2.5. Our goal is to learn a mapping from datapoints in the range of values of $\mathcal{B}(\boldsymbol{\beta}, \boldsymbol{\theta})$ to an abstract representation $\mathbf{z} \in \mathcal{M}$, where $\mathcal{M}$ is a learned manifold. This mapping is essentially an **encoding** by a learned function $\mathcal{E} : \mathbf{x} = \mathcal{B}(\boldsymbol{\beta}, \boldsymbol{\theta}) \rightarrow \mathcal{M}$. Typically, both $\mathbf{x}$ and $\mathbf{z}$ are high-dimensional vectors, where $\mathbf{z}$ serves as a **vector embedding** of $\mathbf{x}$. The mapping function $\mathcal{E}$ is trained such that $\mathbf{z}$ exhibits certain desirable properties. Common objectives include ensuring that $\mathbf{z}$ lies in a lower-dimensional space than $\mathbf{x}$, that its distribution $p(\mathbf{z})$ possesses a simple and well-behaved structure (e.g., a unit normal distribution), and that the dimensions of $\mathbf{z}$ correspond to independent factors of variation. In this manner, the representational space $\mathcal{M}$ becomes simpler, more abstract, and better organized than the original data space $\mathcal{X}$—as conceptually illustrated in Figure 2.5.

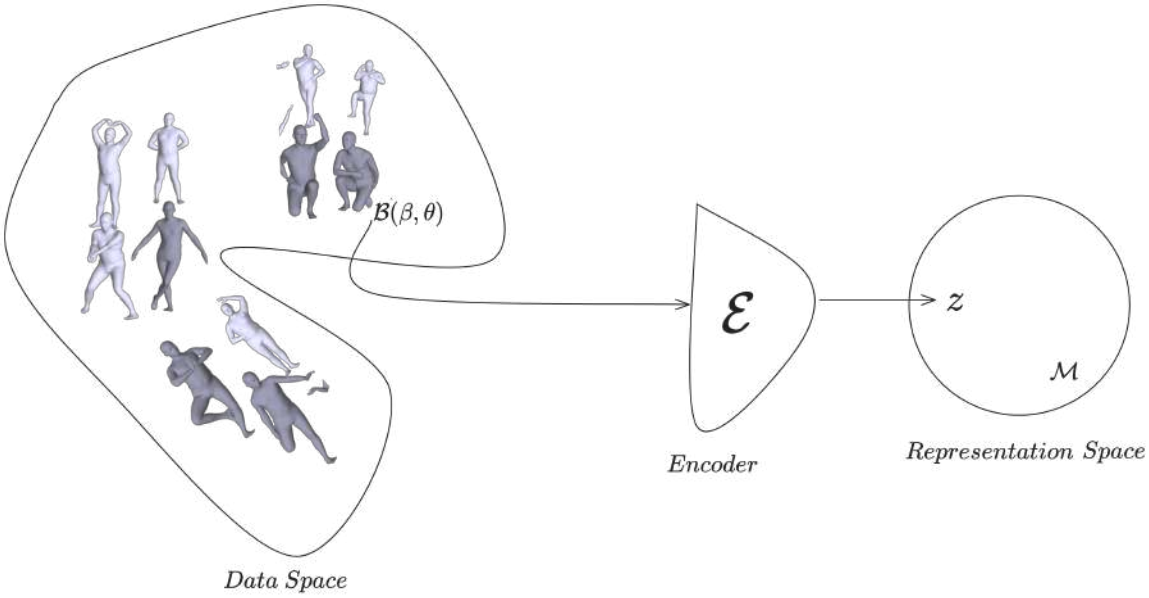


Figure 2.5 Mapping of a pose from Data Space to a compact lower dimension representation in a learned manifold $\mathcal{M}$.

However, a proper representation should have certain properties to be useful, as defined in [62]. First, it should be continuous (i.e, similar poses to be placed in the same region of the $\mathcal{M}$), the learned $\mathcal{M}$ to have certain geometric properties, and to be interpretable. The most common kind of representation learner is called **autoencoder**, A, [62], consisting of an **encoder** $\mathcal{E}$ and a **decoder** $\mathcal{G}$. An autoencoder is a function that maps data back to itself (hence the "auto"), but via a low-dimensional representational bottleneck, as shown in Figure 2.6.

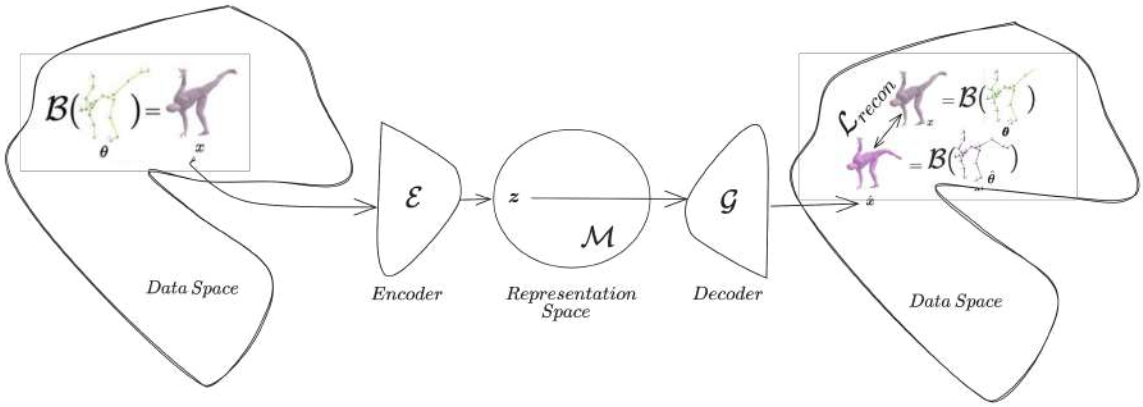


Figure 2.6 Autoencoder architecture for human pose representation. The encoder maps a pose from the high-dimensional data space to a compact latent code $\mathbf{z}$ on the manifold $\mathcal{M}$, and the decoder reconstructs the original pose from $\mathbf{z}$.

One can argue that this might not be particularly useful, as the output of the autoencoder is identical to the input. However, while training an autoencoder, additional constraints are imposed on the intermediate representation $\mathbf{z}$, so that it has properties useful for each specific task. For instance, the most common constraint is compression [62]: $\mathbf{z}$ is a low-dimensional, compressed representation of $\mathbf{x}$. As mentioned, an autoencoder A consists of two main parts, an **encoder** $\mathcal{E}$ and a **decoder** $\mathcal{G}$, with $A = \mathcal{G} \circ \mathcal{E}$. The encoder, $\mathcal{E} : \mathbb{R}^D \to \mathbb{R}^d$, maps high-dimensional human pose data $\mathbf{x} \in \mathbb{R}^D$ to a vector embedding $\mathbf{z} \in \mathbb{R}^d$. Typically, the key property is that $d < D$, that is, we have performed **dimensionality reduction**, and now we can work in a lower-dimensional structured space. There are cases where this property of autoencoders is not needed. Still, there are other properties one might like to enforce on the autoencoder as explicit or implicit constraints on $\mathbf{z}$. On the other hand, the decoder, $\mathcal{G} : \mathbb{R}^d \to \mathbb{R}^D$ performs the inverse mapping to $\mathcal{E}$, and ideally $\mathcal{G}$ is exactly the inverse function $\mathcal{E}^{-1}$. Although it is impossible to identically invert $\mathcal{E}$, a loss function is employed to penalize how far we are from the input, and a common choice is the squared error [62] **reconstruction loss**,

$$\mathcal{L}_{\text{recon}}(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 . \tag{2.3}$$

This loss penalizes the squared Euclidean distance between the original input $\mathbf{x}$ and its reconstruction $\hat{\mathbf{x}} = \mathcal{G}(\mathcal{E}(x))$, encouraging the autoencoder to preserve the information content of $\mathbf{x}$ through the bottleneck. The complete learning objective is then formulated as:

$$\mathcal{E}^*, \mathcal{G}^* = \arg\min_{\mathcal{E},\mathcal{G}} \mathbb{E}_{\mathbf{x}} \left[\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 \right] . \tag{2.4}$$

Essentially, an autoencoder learns a compact, lower-dimensional representation from which the original inputs can be reconstructed with minimal loss, allowing redundant features to be discarded while preserving essential information. By tailoring the training objective, we can promote different properties in the learned representation—such as high-fidelity reconstruction, the ability to generate novel samples, and control over the geometry of the underlying manifold $\mathcal{M}$. In this Thesis, we show that learning such a representation with the right properties is crucial for applications, including motion capture from sparse sensors, mitigating bias in human datasets, or implicitly smoothing the noisy AI estimates.

2.3 Body Fitting

Recovering human body motion from sensor data can be framed as an optimization problem. In this paradigm, a template mesh $\bar{T}$ is fitted to a set of observations that act

as constraints. These constraints guide the optimization algorithm to solve for the unknown body model parameters $(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{T})$ with $\mathbf{T} \in \mathbb{SE}(3)$, that best explain the data. The nature of these constraints depends directly on the capture technology, typically being 3D marker positions for optical MoCap or 2D keypoint detections and silhouette masks for markerless MoCap from images. In most cases, fitting SMPL is usually formulated as minimizing a multi-term objective function, comprising weighted data $\mathcal{E}_{data}$ and prior $\mathcal{E}_{prior}$ terms:

$$\underset{\boldsymbol{\theta},\boldsymbol{\beta},\mathbf{T}}{\operatorname{argmin}} \mathcal{E}_{data} + \mathcal{E}_{prior}, \tag{2.5}$$

where $\boldsymbol{\theta}$ are pose parameters, $\boldsymbol{\beta}$ are shape parameters, and $\mathbf{T}$ the rigid root transformation. The data term $\mathcal{E}_{data}$ measures misfit to observations, and the prior term $\mathcal{E}_{prior}$ regularizes the solution by ensuring that the new pose is near a learned prior from an autoencoding process such as the one described in Section 2.2.

The data term varies depending on the available observations. For **2D keypoints**, a reprojection loss is commonly used—e.g., the robust distance between detected 2D joints and the projection of the model's 3D joints into the image plane [63]. Detection confidence can be used to weight each keypoint's contribution, and robust error functions such as Geman–McClure or Huber loss can down-weight outliers arising from imperfect detections. For **3D points**, $\mathcal{E}_{data}$ can include Euclidean distance errors between model joints or vertices and observed 3D positions. For **silhouettes**, a common choice is to minimize a differentiable overlap loss (e.g., Intersection over Union (IoU)) or a distance field error between the model's rendered mask and the observed image silhouette [64, 65]. Additional terms may include alignment of limb orientations to Part Orientation Fields (POFs) or matching DensePose correspondences by sampling model surface points to image pixels. Each data sub-term is usually scaled by a weight that balances its influence in the overall objective. However, because fitting from sparse data and noisy data can produce erroneous results, human pose priors are required to avoid unrealistic solutions. SMPLify [63] introduced a **pose prior** based on a Gaussian Mixture Model (GMM) of joint angles learned from motion capture datasets, which discourages improbable poses such as excessive bending or twisting. It also uses a **shape prior**, typically a Gaussian on the shape coefficients, to keep body shapes within a realistic human range. More recent work replaces the GMM with learned deep priors: for example, SMPLify-X [4] uses VPoser, a variational autoencoder trained on large human pose datasets, to better capture plausible pose distributions. Another important change is that instead of directly optimizing $\boldsymbol{\theta}$, they optimize the $\mathbf{z}$ lower-dimensional pose space described. Additional priors can penalize self-intersections or collisions, using either precomputed collision proxy bodies or distance field penalties [66, 67]. The fitting is solved by adjusting

$(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T})$ to minimize $\mathcal{E}_{data}$ and $\mathcal{E}_{prior}$. This is a nonconvex, high-dimensional optimization problem. In practice, a multi-stage approach is often used: for example, first optimize the global orientation and limb poses with the shape fixed (often shape set to average), then refine shape with pose fixed, and alternate. This mitigates the entanglement between pose and shape, which can otherwise confuse the solver. Annealing schemes are also employed (e.g., starting with high weights on the prior term and gradually relaxing them as the fit improves) [4]. This ensures the solution stays in a plausible basin initially.

Apart from the constraints, another important pillar in the template mesh fitting process is the appropriate selection of an optimizer. Because SMPL is differentiable (analytic gradients of vertices and joints concerning parameters are available), gradient-based optimizers are the workhorse for SMPL fitting. The most common methods include second-order or quasi-Newton techniques and first-order methods from deep learning, like Adam or SGD. This quasi-Newton optimizer [68] is widely used in SMPL fitting for its efficiency on smooth problems. SMPLify and its successors employ L-BFGS to minimize the objective, often achieving good fits in tens of iterations [63]. For example, SMPLify-X (fitting SMPL-X to image keypoints) is implemented in PyTorch using the L-BFGS optimizer with a strong Wolfe line search. L-BFGS leverages an approximation of the Hessian, which speeds up convergence compared to simple gradient descent. However, it assumes a reasonably well-behaved error surface; with poor initialization, L-BFGS can converge to undesirable local minima (e.g., a mirrored pose that still fits the 2D keypoints). To improve robustness, practitioners may run the optimizer from multiple starting points or use stepped strategies. In PyTorch, a single subject is fitted at a time (batch size 1) with L-BFGS due to its design. Gradient-based methods can directly exploit the known derivatives of the SMPL model and the objective. They tend to converge quickly when the objective is smooth, and the initialization is within the attraction basin of the correct solution. These methods are also memory-efficient (L-BFGS requires storing only a limited history), and can seamlessly incorporate complex losses such as differentiable rendering, thanks to automatic differentiation in frameworks like PyTorch. However, they are sensitive to initialization and prone to local minima. Ambiguities in monocular 2D data can lead to plausible-but-wrong poses (e.g., flipped orientation), and strong priors, while helpful, do not fully prevent such errors. Flat regions in the loss landscape (e.g., silhouette losses insensitive to in-plane rotations) can cause the optimizer to stall. Gradient-based fitting can also be sensitive to noisy data; outlier observations, such as mis-detected keypoints, may pull the solution away from the optimum. Robust loss functions alleviate this issue but require careful tuning. Apart from that, all the above calculate the gradients in the Euclidean space, which is not the case for $\boldsymbol{\theta}$, which lies on the $\mathbb{SO}^3$ manifold, or in the case of optimizing the la-

tent $\boldsymbol{\theta}$ on a hyperdimensional $\mathcal{M}$ with an unknown geometric structure. Within the next chapters of this Thesis, we will exploit the power of representation learning to improve the optimization process, but also indicate the importance of directly optimizing on the manifold space, which yields significant improvements, both in terms of convergence and quality.

2.3.1 Markers

In professional optical motion capture systems, a set of retro-reflective markers is placed on an actor's body, and their 3D positions are recorded with high precision by multiple calibrated cameras, or in our case, with low-cost depth devices and inferred using AI. To recover the body pose from this data, we define an objective function that minimizes the Euclidean distance between the 3D positions of the observed markers and the corresponding virtual marker locations on the surface of the body model. Given a set of N_m observed 3D marker positions $\{\boldsymbol{\ell}_{m,i} \in \mathbb{R}^3\}_{i=1}^{N_m}$ and a function $\mathcal{J}_i(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{T})$ that returns the 3D location of the i-th virtual marker on the SMPL mesh under shape parameters $\boldsymbol{\beta}$, pose parameters $\boldsymbol{\theta}$, and global rigid transformation $\mathbf{T}$, the marker-based fitting term, $\mathcal{E}_{\text{marker}}$, is formulated as a sum of squared L2 distances:

$$\mathcal{E}_{\text{marker}}(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{T}) = \sum_{i=1}^{N_m} \|\boldsymbol{\ell}_{m,i} - \mathcal{J}_i(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{T})\|_2^2 \,. \tag{2.6}$$

This objective is then minimized with respect to the shape, pose, and global root transform parameters $\boldsymbol{\beta}$, $\boldsymbol{\theta}$ and $\mathbf{T}$, to align the model with the captured marker data.

2.3.2 2D keypoints

With the advent of deep learning, markerless motion capture from monocular RGB video has become highly accessible. State-of-the-art 2D human pose estimation networks, such as OpenPose [69] or the most recent Sapiens [70], can accurately detect the 2D pixel locations of anatomical keypoints (e.g., wrists, elbows, knees) in an image. These 2D detections can serve as powerful constraints for fitting a 3D body model.

The fitting process involves projecting the 3D joint locations of the body model onto the 2D image plane and minimizing the difference between these projected points and the detected 2D keypoints. Similarly with before, $\mathcal{J}_i(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{T})$ is the function that returns the 3D locations of the model's joints. Let $\boldsymbol{\pi}(\cdot)$ be the camera projection function that maps a 3D point to 2D pixel coordinates. Given a set of K detected

2D keypoints $\{\mathbf{k}_j \in \mathbb{R}^2\}_{j=1}^K$ with corresponding detection confidences $\{c_j \in [0,1]\}_{j=1}^K$, the 2D reprojection error, $\mathcal{E}_{2\text{D}}$, is formulated as a weighted sum of squared distances:

$$\mathcal{E}_{2\text{D}}(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{T}, \boldsymbol{\Pi}) = \sum_{j=1}^{K} c_j \left\| \mathbf{k}_j - \boldsymbol{\pi}\left(\mathcal{J}_j(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{T}); \boldsymbol{\Pi}\right) \right\|_2^2 . \tag{2.7}$$

Here, $\boldsymbol{\Pi}$ represents the camera parameters. This objective forces the model to conform to the 2D evidence, but as we will see, it requires strong priors to resolve the uncertainties, and high-level noise introduced by the AI estimators.

2.4 Human Pose Priors

This Thesis addresses the challenge of recovering whole-body pose and motion from partial, noisy, and incomplete data. Such data is inherent to common modalities; for example, optical motion capture is affected by sensor noise and marker occlusions, while AI-driven markerless systems often produce keypoint mis-detections, missed detections, or high-frequency jitter (see Section 1.2). To overcome these issues, we aim to leverage a learned representation of human poses. This approach projects poses onto a learned latent manifold where all plausible human configurations reside. Operating within this constrained space allows us to regularize the estimation problem by applying priors to the body model parameters. This ensures that the final estimated poses are both natural and physically plausible. Given that the space of human poses is high-dimensional and notoriously difficult to model directly, this section details various priors designed to navigate this complexity.

The most intuitive way to model a pose prior is via a GMM on the axis-angle representation of body pose, as famously used in SMPLify [63]. This approach clusters a database of real human poses and models the distribution as a weighted sum of Gaussians. However, a GMM presumes that the target space can be well-approximated by a linear combination of unimodal Gaussian distributions. This may be too strong an assumption for the complex, non-linear manifold of natural body poses. To address this limitation, recent work has leveraged deep learning to create more expressive priors:

- **Adversarial Priors:** Used in HMR [71], this technique employs a discriminator network trained in a Generative Adversarial Network (GAN) framework. The discriminator learns to distinguish between poses from a real motion capture dataset and those generated by the pose estimation network. Its feedback is used as a loss to penalize poses that are identified as unrealistic.

- **VAE-based Priors:** VPoser [4] utilizes a Variational Autoencoder (VAE) to learn a holistic prior over the entire body pose. The VAE maps high-dimensional SMPL pose vectors into a low-dimensional, continuous latent space, where the distribution is encouraged to follow a simple Gaussian. By sampling from this latent space and decoding the samples, VPoser can generate a wide variety of realistic human poses. This provides a powerful prior that forces estimated poses to lie on the learned manifold of plausibility.

- **Normalizing Flows:** Normalizing Flows [72] offers an even more flexible method for modeling complex probability distributions. They construct a complex distribution by transforming a simple base distribution (e.g., a standard normal) through a sequence of invertible and differentiable mappings. This allows for the exact computation of the probability density for a given pose, providing a highly expressive and powerful prior.

- **NRDF-based Priors:** Neural Residual Density Fields (NRDF) [73] are used to model the pose prior as a continuous energy or density field defined over the high-dimensional human pose space. Instead of relying on a latent bottleneck (as in VAEs) or strictly invertible transformations (as in flows), NRDF learns a residual correction over a base distribution, enabling flexible modeling of complex, multi-modal pose manifolds. The learned field provides a differentiable likelihood (or energy) for any input pose, which can be directly integrated into optimization-based fitting frameworks.

- **Discrete Latent Priors (Vector-Quantized Models):** Recent work, such as [74], models human pose and shape in a discrete latent space using vector-quantization mechanisms. Instead of learning a continuous Gaussian latent representation (as in VAEs), these methods encode high-dimensional pose parameters (e.g., SMPL pose $\boldsymbol{\theta}$ and shape $\boldsymbol{\beta}$) into a finite set of learned codebook embeddings. Formally, an encoder maps pose parameters to a latent vector $\mathbf{z}$, which is then quantized to the nearest codebook entry from a learned dictionary. The decoder reconstructs the pose from this discrete representation. By constraining solutions to lie within a structured set of learned pose tokens, vector-quantized models implicitly regularize the solution space and improve stability in pose and shape estimation. Unlike diffusion models, these approaches do not rely on iterative denoising, and unlike normalizing flows, they do not require invertible mappings, making them computationally efficient while still capturing complex pose distributions.

- **Diffusion-Based Priors:** Diffusion models have recently been adopted to learn rich human pose priors by modeling the distribution of plausible 3D poses via

iterative denoising. Works such as DPoser [75] demonstrate that diffusion can serve as a generative prior, improving pose realism and robustness across tasks such as pose completion, mesh recovery, and denoising. Extensions and variants in 2025 further refine these ideas by incorporating hierarchical temporal pruning [76] and autoregressive spatial-temporal guidance [77], addressing efficiency and temporal coherence in generated pose sequences. Other diffusion approaches investigate discrete diffusion processes for occluded pose estimation [78] or multimodal conditioning to boost generalization in pose estimation and completion [79, 80]. Unlike continuous latent priors (VAEs) or codebook-based discrete priors (VQ models), diffusion methods provide flexible multimodal distributions without imposing strict invertibility constraints, making them expressive and powerful generative priors for human pose modeling. However, their iterative denoising approach is computationally inefficient, limiting their applications in real-time scenarios.

Still, a key limitation of most existing human pose priors is that they operate on the latent space without considering the geometric structure of the learned **manifold** $\mathcal{M}$. Consequently, they fail to leverage its intrinsic properties. By contrast, explicitly modeling the manifold's geometry—for instance, by constraining it to be a **hypersphere**—unlocks significant advantages. A hyperspherical structure enables principled interpolation between poses, facilitates direct optimization on the manifold for superior **convergence**, and ultimately yields higher-quality results. In the following chapters, we demonstrate how this simple yet powerful hyperspherical representation significantly enhances the quality of motion capture analysis.

2.5 Interpolation in Latent Space

Another important concept used within this Thesis is the interpolation in latent space. The development of powerful, deep-learning-based pose priors, particularly VAEs, such as VPoser [4], does more than regularize optimization. By learning a mapping from the high-dimensional pose space to a compact and well-behaved latent space, they enable meaningful semantic operations such as interpolation. Generating smooth and plausible transitions between two distinct poses, a non-trivial task in the original axis-angle space, becomes a simple linear operation in the learned latent space. Suppose we have two poses, $\boldsymbol{\theta}_A$ and $\boldsymbol{\theta}_B$, which are encoded by a VAE into latent vectors $\mathbf{z}_A$ and $\mathbf{z}_B$, respectively. A smooth and realistic motion sequence transitioning from pose $\boldsymbol{\theta}_A$ to pose $\boldsymbol{\theta}_B$ can be generated by interpolating between their latent representations and then decoding the intermediate vectors back into full-body poses.

For a standard Linear Interpolation (LERP), the intermediate latent vector $\mathbf{z}_{\text{int}}$ at time $t \in [0, 1]$ is given by:

$$\mathbf{z}_{\text{int}}(t) = (1 - t)\mathbf{z}_A + t\mathbf{z}_B. \tag{2.8}$$

For latent spaces where direction is more important than magnitude, or to ensure constant-speed transitions, Spherical Linear Interpolation (SLERP) is often preferred:

$$\mathbf{z}_{\text{int}}(t) = \frac{\sin((1 - t)\psi)}{\sin(\psi)}\mathbf{z}_A + \frac{\sin(t\psi)}{\sin(\psi)}\mathbf{z}_B, \tag{2.9}$$

where $\psi = \arccos(\mathbf{z}_A \cdot \mathbf{z}_B)$ is the angle between the two latent vectors, assuming that they are unit-normalised. This ability to generate, edit, and transition between poses via simple latent space arithmetic is fundamental for the applications in motion synthesis and completion that are explored in this Thesis.

A central argument of this Thesis is that the quality of the resulting motion is significantly impacted by two key elements: the intrinsic geometry of the learned manifold and the mathematical properties of the interpolation scheme used to navigate it. We establish that higher-order interpolation methods not only improve the generation of new samples by affording greater variability but also produce smoother transitions. This effect is magnified when operating on a manifold with well-defined geometric properties, which inherently enforces temporal consistency across pose sequences.

Chapter 3

Related Work

In this chapter, we provide a comprehensive literature review. We begin by discussing data balancing techniques, which are crucial for training robust models on imbalanced datasets. We, then, explore model-based optimization methods, focusing on their application in human pose estimation and motion capture. We also delve into temporal constraints, highlighting their importance in ensuring smooth and consistent motion sequences. Finally, we examine the role of learned pose manifolds in enhancing the realism and plausibility of generated human motion. Through this review, we aim to establish a solid foundation for the methodologies and approaches presented in this Thesis.

3.1 Data Balancing

The effectiveness of any AI model is inherently tied to the quality and distribution of its training data. In human motion modeling, the available datasets are limited and often exhibit notable shortcomings. For instance, large collections such as AMASS [2] and Fit3D [81] contain considerable redundancy and display a strong long-tail distribution, while synthetic ones such as BEDLAM [82] suffer from the problem of domain-gap. Consequently, rare or extreme poses are underrepresented and difficult for regression networks to learn—an issue amplified by stochastic mini-batch optimization and the bias introduced by the chosen estimators.

Several research efforts have addressed this imbalance problem. Some methods are specifically designed for motion or pose data, such as [83], which introduces a prototype-classifier branch to initialize iterative refinements, while others borrow ideas from imbalanced classification and adapt them to regression. Most existing

solutions fall into two broad categories: *re-sampling* and *re-weighting*. Re-sampling strategies aim to modify the training distribution, either by undersampling frequent samples [84], oversampling rare ones through interpolation [85], perturbation with controlled noise [86], or by combining these operations in hybrid forms [87]. However, when the data consists of high-dimensional, structured outputs—such as articulated human poses—defining meaningful interpolation paths or identifying "rare" instances becomes non-trivial, making standard re-sampling less applicable.

An alternative family of approaches, often referred to as *utility-based* or *cost-sensitive* regression [88], assigns relevance weights to training samples according to a user-defined utility function. This weighting principle is also foundational to several re-sampling methods for regression [84]. More recent work extends these ideas by using kernel density estimation [89], adapting evaluation metrics into differentiable losses [90], or employing feature/label smoothing and binning schemes [91].

A different direction is emerging through *contrastive* or *pairwise* formulations. For instance, RankSim [92] regularizes training to enforce proximity in both feature and prediction spaces, while BMSE [11] introduces a contrastive-like objective using intra-batch minimum-error sample classification. BMSE defines a cross-entropy loss based on the L_2 reconstruction error, interpreted probabilistically as a likelihood term. Yet, most of these methods depend on discretizing or binning the output space—an operation that is inherently challenging for continuous, high-dimensional signals like full-body human pose.

3.2 Model-Based Optimization

The most prominent method in estimating motion from sparse constraints involves fitting a low-dimensional parametric body model, like SCAPE [29] or SMPL [27] or the recently introduced MHR [42], to 2D observations. These models can generate a realistic human mesh from a small set of pose parameters (controlling joint articulation) and shape parameters (controlling identity-specific body proportions). Approaches to estimate these model parameters from an image can be broadly categorized into *optimization-based*, *regression-based*, and *hybrid methods*. Early works in the former used silhouettes for fitting the template mesh [93, 94]. However, with the advent of robust 2D pose estimators like OpenPose [69], the standard became minimizing the reprojection error between the 2D-projected joints of the 3D model and the detected 2D keypoints. The seminal work in this area is SMPLify [63], which fits the SMPL model to 2D keypoints detected in an image. To prevent unnatural poses and ensure anthropometrically plausible body shapes, the objective function includes powerful

pose priors (e.g., a variational autoencoder over a large motion capture dataset) and shape priors. While these methods can achieve high-fidelity alignment to 2D evidence without needing 3D supervision for training; they have notable drawbacks. The objective function is highly non-convex, making the iterative optimization process slow and prone to getting stuck in local minima. The final result is susceptible to the quality of the 2D keypoint detections and the initialization of the parameters. Subsequent research has focused on improving initialization with image-based cues [95], extending the framework to multi-view [96] and multi-person scenarios [97, 98], and integrating physics for more dynamic and realistic results [99]. The latter methods directly regress templates' mesh parameters. Human Mesh Recovery (HMR) [71] uses a convolutional neural network (CNN) to predict SMPL parameters directly from an RGB image. A key challenge for regression methods is the scarcity of in-the-wild images with corresponding 3D ground truth meshes. HMR cleverly circumvented this by employing an adversarial training scheme. A discriminator is trained to distinguish between regressed parameters and parameters from a real 3D dataset (e.g., MoCap data), forcing the regressor to produce plausible human poses and shapes. This allows the model to be trained using a combination of datasets with 2D and 3D labels, and even images with no labels at all. Following HMR, numerous architectural improvements were proposed. GraphCMR [100] introduced graph convolutions to better model the relationships between mesh vertices. More recently, Transformer-based architectures like METRO [101, 102] have shown significant promise by treating joints and mesh vertices as tokens, enabling the model to learn non-local interactions for a more holistic understanding of the body. Regression-based methods [103–105] are extremely fast, requiring only a single forward pass through the network. However, they can sometimes produce results that are less precisely aligned with the 2D image evidence compared to optimization-based methods. The last methods try to combine the benefits of the aforementioned methods. A prominent example is SPIN (SMPL oPtimization-IN-the-loop) [95]. SPIN first uses a neural network to regress an initial set of SMPL parameters. These parameters are then refined using a few optimization steps that minimize a traditional SMPLify-style objective function. Crucially, this entire process is differentiable, allowing the regression network to be trained with self-supervision signals derived from the optimization outcome. The network learns to predict better initializations that lead to lower objective errors after refinement, effectively learning from its own mistakes. This "in-the-loop" strategy has proven highly effective and has been adopted by many subsequent state-of-the-art methods. More recent monocular methods [106–108] exploited SLAM to estimate camera parameters, resolving ambiguities, while [109] estimates human poses in a novel Gravity-View coordinate system, defined by the world gravity and the camera view direction. HumanMM [110] introduced a human motion prior to reconstructing

long-sequence 3D human motion in the world coordinates, while [111, 112] combined additional information like spatial and semantic prompts, camera calibration, and shape estimation to improve performance.

3.2.1 Manifold Optimization

The optimization of functions defined on non-Euclidean spaces is a long-standing problem that has found renewed importance in modern machine learning and computer vision. By operating directly within the manifold's intrinsic geometry, manifold optimization methods can offer significant advantages in terms of efficiency, stability, and the enforcement of inherent constraints. These methods generalize traditional unconstrained optimization algorithms from Euclidean space $\mathbb{R}^n$ to smooth manifolds $\mathcal{M}$. The foundational framework for many of these techniques was extensively detailed in the seminal work of Absil, Mahony, and Sepulchre [113], which established the necessary geometric concepts like tangent spaces, retractions, and vector transport. Early efforts in this domain logically focused on extending the most fundamental first-order algorithm: gradient descent. Riemannian Gradient Descent (RGD) replaces the standard linear update with a step along the manifold. The search direction is determined by the Riemannian gradient $gradf(\mathbf{z})$, which is a vector in the tangent space $T_{\mathbf{z}}\mathcal{M}$ at the current iterate $\mathbf{z}$. The update, then, proceeds by moving from $\mathbf{z}$ in this direction. The classical formulation of this step is via the exponential map, which moves along a geodesic—the generalization of a straight line to a curved space. The update rule is given by:

$$\mathbf{z}_{k+1} = exp_{\mathbf{z}_k}(-a_k gradf(\mathbf{z}_k)), \tag{3.1}$$

where a_k is the step size. Several works have explored the global convergence properties and practical applications of geodesic-based gradient descent [114, 115]. However, computing geodesics is often computationally expensive or lacks a closed-form expression for many manifolds of interest. This practical challenge led to the development of retractions, which are more general and computationally cheaper first-order approximations of the exponential map [113]. A retraction $R_{\mathbf{z}_k} : T_{\mathbf{z}_k} \to \mathcal{M}$ provides a feasible update on the manifold without the cost of computing the exact geodesic. This concept, formalized by Ring and Wirth [116], has become central to the design of practical manifold optimization algorithms. Building on first-order methods, researchers have also developed more sophisticated Riemannian counterparts to second-order techniques. Riemannian trust-region methods, for example, solve a sequence of subproblems within a "trust region" defined on the tangent space, offering robust convergence guarantees [117]. To avoid the explicit computation of the

Riemannian Hessian, Riemannian quasi-Newton methods have been proposed. These methods, particularly the Riemannian Broyden-Fletcher-Goldfarb-Shanno (R-BFGS) algorithm, incrementally build an approximation of the Hessian using only first-order information. A key challenge is that the gradients (and the Hessian approximation matrix) at different iterates reside in different tangent spaces. To address this, a vector transport operation is used to move the previous Hessian approximation into the current tangent space before applying the BFGS update. The work by Qi et al. [118] presents a robust R-BFGS framework that relies on general retractions and vector transports, making it highly flexible and efficient. More recent works have further refined the convergence theory and practical performance of these methods [119].

Such approaches are particularly valuable for human body fitting, as they mitigate the distortions that can arise when projecting high-dimensional or curved data onto a Euclidean space. They also enhance both optimization stability and accuracy. To this end, we propose a manifold-based optimization framework coupled with a VAE that embeds human poses onto a high-dimensional hypersphere, demonstrating its efficacy in reconstructing human motion from sparse multi-view videos.

3.3 Temporal Constraints

Per-frame pose fitting can suffer from severe temporal jitter when estimates are noisy. Although temporal filtering can suppress such inconsistency [120], it is notoriously difficult to design filters that handle occlusions or outliers effectively. Consequently, most modern approaches incorporate temporal priors or explicit smoothness objectives into the optimization process.

An early attempt in this direction is [121], which used small bundles of input frames to jointly regress coherent SMPL parameters. Subsequent work [122] adopted recurrent architectures that naturally encode temporal dependencies, allowing each prediction to depend on preceding frames. Within optimization-based frameworks, the most common temporal objective is joint-position smoothness [6, 9, 123, 124], enforcing local continuity of motion. A discrete cosine transform (DCT) prior has also been used to impose smoothness when jointly fitting SMPL to image sequences [5]. Higher-order terms such as velocity or acceleration constancy [1, 2, 125] are another alternative, applied to either joint positions or angular representations.

With the rise of learned priors, many studies complement joint-based constraints with smoothness in higher-level latent spaces—for instance, the feature space of velocity autoencoders [66, 126], pose latents [8], motion embeddings [127], or autoregressive

transition states [9, 128]. A unifying characteristic of these methods is their reliance on additional objectives that introduce new hyperparameters requiring careful tuning—an especially complex task when numerous losses coexist in a body-model fitting pipeline. In contrast, our method achieves temporal coherence *implicitly* by smoothly interpolating along a learned pose manifold, removing the need for extra smoothing terms or hyperparameter adjustment.

3.4 Human Pose Priors

Learning human pose priors has become a cornerstone for addressing ambiguity, occlusions, and incomplete observations in pose or motion estimation. These priors typically aim to learn the unconditional distribution $p_{\text{data}}(\mathbf{x})$ of plausible poses and to represent them on a low-dimensional manifold [129]. [63] introduced SMPLify, fitting a Gaussian mixture model to MoCap data, while [4] proposed the VAE-based VPoser for encoding 3D poses into a latent space. Adversarial alternatives such as [130] employ GANs to discriminate between realistic and generated poses, and PoseNDF [131] learns pose manifolds as neural implicit fields.

Other approaches define *conditional* priors $p_{\text{data}}(\mathbf{x}|cond)$ tailored to specific contexts. MVAE [132] and HuMoR [99] learn autoregressive conditional VAEs that model plausible motion trajectories given past frames, while ACTOR [133] introduces an action-conditioned variational prior using a Transformer-based VAE. HMR [71] and VIBE [134] combine reconstruction and adversarial losses in joint training, whereas GF-Pose [135] employs a score-based diffusion model with hierarchical conditioning to unify task-dependent and task-independent priors.

Despite their variety, most of these methods assume sampling from uniform or Gaussian latent distributions, implicitly treating the manifold as a Euclidean or unit-sphere space. We show that explicitly enforcing a spherical geometry strengthens interpolation behavior—an essential property for bundle keyframe solving, where latent continuity directly affects motion quality. To this end, we introduce **SPoser**, a spherical autoencoder that maps human poses onto a hypersphere, enabling geometrically consistent and higher-order interpolation across latent representations.

3.4.1 Geometry-Aware Manifolds

Variational Autoencoders (VAEs) [136] map complex data into a lower-dimensional latent space constrained by a prior distribution. Most implementations adopt Gaus-

sian priors for both posterior and prior distributions, a simplification that disregards the latent space's underlying geometry [137]. This mismatch can hinder reconstruction fidelity and generate discontinuous interpolations where manifold assumptions no longer hold. Recent research addresses this by exploring non-Euclidean latent topologies, including hyperspherical [138, 139], toroidal [140], or Riemannian manifolds [141]. While the von Mises–Fisher (vMF) distribution used by [138, 139] better models directional data, it still inherits the limitations of variational inference. [142] alleviated this by directly applying spherical normalization to latent variables, avoiding probabilistic approximations altogether. Building on this insight, we propose **SPoser**, a spherical pose prior that encodes plausible human configurations on an n-dimensional hypersphere.

3.4.2 Latent Space Interpolation

Interpolation on the unit sphere S^2 is defined for ordered points $\{\mathbf{p}_i : \|\mathbf{p}_i\| = 1, i = 0, \ldots, n\}$ on S^2, producing a curve $\mathbf{p}(t) \in S^2$ such that $\mathbf{p}(i) = \mathbf{p}_i$ for uniformly spaced $t = i$. Spherical interpolation is widely used in computer graphics [143] and robotics [144] due to its geometric interpretability. Among the most common schemes are spherical linear interpolation (SLERP) and spherical quadrangle interpolation (SQUAD), providing C^0 and C^2 continuity, respectively. [145] recently proposed Spherical Interpolation of DER-n (SIDER-n), a higher-order generalization capable of producing smooth C^n interpolants directly on S^2.

Latent interpolation has also been employed in generative modeling. [146] demonstrated that interpolating latent vectors in image-space VAEs can produce controllable semantic transitions, while [147] applied a similar idea to sentence representations in RNN-based language models. In human motion modeling, [148] interpolated pose latents to synthesize novel samples for evaluating prior quality. However, ensuring continuity and smoothness in latent space remains challenging: discontinuities or "dead zones" can lead to abrupt, unrealistic transitions. Architectural improvements and regularization strategies have been proposed to encourage more uniform latent manifolds [149]. Nevertheless, the majority of existing work still relies on first-order interpolation (SLERP). Here, we adapt the SIDER-n formulation to interpolate between latent vectors $\mathbf{z}$ on the learned manifold $\mathcal{M}$, enabling higher-order smoothness and more expressive motion reconstruction.

3.5 Bundle Solving

Temporal consistency is more effectively enforced when jointly solving for bundles of consecutive frames rather than individual ones. For instance, [5] introduced a DCT prior at the second stage of optimization when jointly fitting groups of 30 frames. Similarly, MoSculp [124] and other works [9, 123, 127] applied joint-smoothness constraints over small temporal windows. Optimization in the latent space of regressors [123] can also support bundle solving, though at the expense of substantial parameter complexity.

More ambitious methods solve entire sequences simultaneously [6], leveraging full temporal context and incorporating complex objective terms such as human–object interaction [126]. However, these approaches depend on accurate initialization—often obtained through a per-frame regressor [6] or initial body fitting stage [126]—and typically require multiple optimization passes. Such multi-stage pipelines are common both in monocular setups [4, 63] and in multi-view variants [5, 127], where early stages initialize frame-wise fits and later ones enforce temporal smoothness [6, 9, 66, 123, 127]. While effective, these procedures are computationally intensive, as each pass reprocesses the same video data.

In contrast, within this Thesis, we will introduce a formulation based on temporal keyframes split across temporal windows and only a one-time initialization for the first frame. It, then, solves for temporally coherent motion in a single optimization stage, achieving robustness and smoothness without explicit multi-stage refinement.

Chapter 4

Removing Bias from Human Pose Datasets

This thesis chapter originally appeared in the literature as [22]

Broadening the accessibility of MoCap technology requires a substantial reduction in the associated financial barriers and the required expertise to operate such systems. Optical systems, while precise, depend on costly sensor arrays and controlled environments. Moreover, even the data acquired from these high-end setups often contains high-frequency noise and artifacts—such as occlusions or ghost markers—that demand extensive manual post-processing. This issue becomes particularly severe when low-cost sensors are employed, as their degraded measurement quality amplifies the noise and error rates, further restricting the usability of MoCap in broader contexts.

Several studies have attempted to mitigate these problems through data-driven methodologies that exploit existing MoCap repositories. Nevertheless, such approaches are fundamentally constrained by a critical issue: the high degree of redundancy in motion sequences. Since many movements are composed of recurring poses (for instance, idle or standing phases), the resulting imbalance skews learning towards over-represented actions and weakens model performance on rare or complex poses.

In this chapter, we tackle the problem of long-tailed data distributions through a representation learning framework. We introduce an imbalanced regression strategy that operates without any additional annotations or external data. The method automatically detects and reweights rare samples during training by leveraging a Mahalanobis-distance-based relevance measure. At the same time, we employ high-order interpolation techniques to sample the latent space of a VAE more effectively, producing diverse and physically coherent synthetic examples for underrepresented

poses. Our experiments confirm that this framework yields substantial improvements, particularly on rare poses, while maintaining architecture independence—allowing it to integrate seamlessly with diverse learning models.

4.1 Introduction

Even though optical MoCap remains the reference standard in the motion analysis community, it still faces a major obstacle: *extensive manual post-processing*. Tasks such as labeling, ghost marker removal, and trajectory correction are labor-intensive and demand expert supervision. For instance, as it has been reported in [16] that cleaning merely **25 minutes** of captured motion—roughly 29000 frames across nine sequences—required nearly **46 hours** of human effort. This burden escalates with lower-cost setups, where fewer and noisier sensors generate measurements that are both inconsistent and error-prone, drastically increasing cleanup time.

AI and data-driven pipelines have emerged as potential solutions for automating such procedures. However, their deployment has mostly been confined to offline or archival data, rather than real-time scenarios. Existing research has explored multiple directions: direct marker labeling [150, 151], regression-based prediction of motion trajectories [152], skeletal pose estimation and motion solving [12, 153], and cross-representation mappings [154]. Some studies have also examined the influence of sensor noise from consumer-grade systems [12]. Despite these advancements, even state-of-the-art models remain susceptible to both information-driven errors (e.g., occlusions and marker swaps) and measurement-level distortions (e.g., jitter or positional drift).

A deeper challenge arises from the intrinsic properties of MoCap datasets themselves. Variability in marker placement between sessions can severely impair generalization, a limitation that has been implicitly modeled in some works [12, 154] and explicitly addressed in others [153]. Another mitigation strategy involves aligning raw observations to parametric human models for a convenient standardized representation [1, 2, 150–152]. Nonetheless, the dominant difficulty lies in the high **data redundancy** inherent in MoCap repositories. Most sequences feature a majority of common movements such as standing or walking, which induces bias in data-driven training and results in a pronounced **long-tailed distribution** [83]. Traditional techniques—like uniform temporal downsampling—reduce sample counts but fail to alleviate the underlying imbalance.

This exposes a methodological gap: conventional solutions for learning from

imbalanced datasets have been designed mainly for **classification** settings, where discrete labels are available. These strategies cannot be directly transferred to **regression** problems—such as predicting continuous joint coordinates from raw marker data—that dominate MoCap applications.

To overcome these challenges, we propose several complementary innovations:

- A representation-learning-driven approach that simultaneously performs oversampling and balanced regression, addressing both redundancy and long-tailed pose distributions.
- A high-order interpolation mechanism for generating richer and more realistic motion samples, surpassing the expressive limits of traditional interpolation schemes.
- A Mahalanobis-distance-based weighting criterion for automatically identifying and reweighting rare samples during training, offering a more discriminative alternative to reconstruction-based relevance metrics.
- A comprehensive experimental evaluation across multiple datasets, confirming both the accuracy and generalization capabilities of the proposed framework.

In particular, we extend classical latent-space sampling by introducing high-order interpolation methods capable of traversing the VAE's manifold more faithfully than standard Linear (LERP) or Spherical Linear (SLERP) interpolation. The latter often yield implausible poses due to latent discontinuities. By interpolating among multiple control anchors, our approach produces smoother transitions and enhances variability, thereby improving model robustness and generalization—especially in complex motion patterns. Furthermore, adopting the Mahalanobis distance instead of the Euclidean metric allows for a better characterization of correlations among pose dimensions, reflecting realistic joint dependencies. Unlike Euclidean distance, which treats each coordinate independently, the Mahalanobis measure captures the covariance structure of human motion, making it more effective in identifying rare or atypical poses. We validate the advantages of this representation through extensive quantitative and qualitative experiments across diverse datasets, demonstrating its plug-and-play compatibility with different architectures. We also analyze the influence of VAE design on performance and evaluate our approach on datasets featuring rare configurations such as yoga sequences, verifying its ability to generalize beyond common motion categories. Together, these experiments provide a robust foundation for the subsequent chapters of this Thesis.

4.2 Approach

4.2.1 Training Framework

Following prior work [151], we operate on parameters of a parametric body model that synthesizes marker observations. Because these markers are generated synthetically, they can be deliberately augmented and perturbed with artifacts and noise [12, 153, 154]. Figure 4.1 outlines the overall training pipeline. We first introduce our strategy for addressing redundancy and long-tail effects in the data (Section 4.2.2), then describe the design choices for the denoising and joint-solving network (Section 4.2.3), and finally present the specifics of the VAE used to support sampling and relevance estimation (Section 4.2.4).

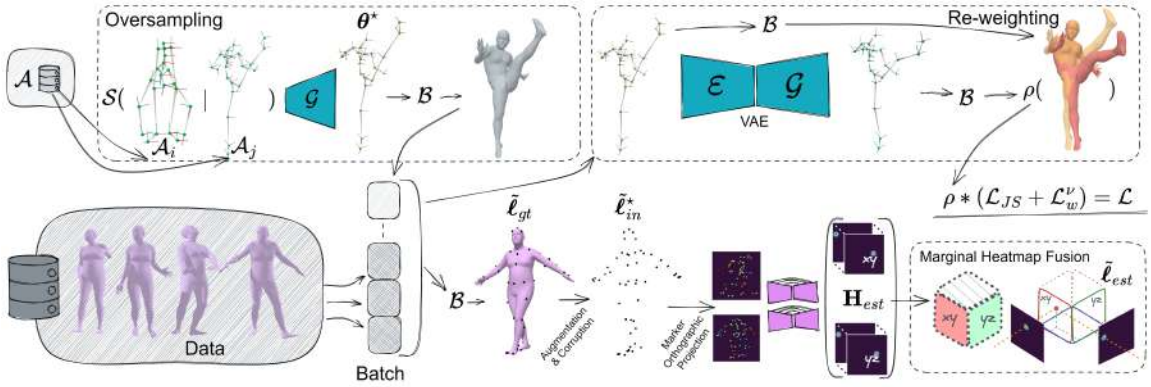


Figure 4.1 Overview of the proposed training pipeline. Starting from an existing motion corpus (*bottom left*), a subset of encoded tail-anchor poses $\mathcal{A}$ is automatically identified through statistical thresholding (Section 4.2.2). These anchors are randomly blended using the sampling operator $\mathcal{S}$ and decoded through the generator $\mathcal{G}$ to synthesize additional rare samples—effectively oversampling the tail distribution during training. A UNet-based model (Section 4.2.3, *bottom middle*) processes two orthographic depth-map projections (xy and yz planes) derived from augmented and noise-corrupted marker positions $\boldsymbol{\ell}^{\star}_{in}$ (originally from $\boldsymbol{\ell}^{\star}_{gt}$) sampled on the body surface $\mathcal{B}$. It outputs two corresponding orthogonal heatmaps, which are marginally fused along the y-axis to recover the 3D landmark estimates $\tilde{\boldsymbol{\ell}}_{est}$ (Section 4.2.3, *bottom right*). For each training batch, the loss contribution of each sample is adaptively scaled by its relevance weight ρ, computed from the Mahalanobis distance of its reconstruction error (Section 4.2.2, *top right*).

4.2.2 Balancing Regression

Utility/relevance functions have long underpinned imbalanced *regression* by steering re-/over-/inter-sample selection and synthesis [84, 85, 87, 88]. Rather than prescribing relevance via hand-crafted rules or closed-form criteria, we learn it directly from data in an unsupervised manner through representation learning. Autoencoding synthesis models [155, 156] jointly learn a reconstruction map and a generative sampler:

$$\boldsymbol{\theta}^{\ddagger} = \mathcal{G}(\mathcal{E}(\boldsymbol{\theta})), \qquad \boldsymbol{\theta}^{\star} = \mathcal{G}(\mathcal{S}(\cdot)), \tag{4.1}$$

subject to constraints on the input $\boldsymbol{\theta}$ and the latent variables $\mathbf{z} = \mathcal{E}(\boldsymbol{\theta}),\ \mathbf{z} \in \mathbb{R}^d$.

An encoder $\mathcal{E}(\boldsymbol{\theta})$ maps $\boldsymbol{\theta}$ into a latent code $\mathbf{z}$, which the generator $\mathcal{G}(\mathbf{z})$ reconstructs into $\boldsymbol{\theta}^{\ddagger}$. Using a sampling operator $\mathcal{S}$ over the latent space, the same model can produce novel samples $\boldsymbol{\theta}^{\star}$. We leverage this hybrid nature to construct an imbalanced-regression solution that (i) oversamples the tail region and (ii) reweights rarer samples during optimization. Our implementation relies on a VAE [155] detailed in Section 4.2.4.

By design, autoencoders reflect the statistics of their training corpus: they reconstruct structures they encounter frequently with higher fidelity, while underrepresented patterns tend to be reconstructed less accurately [157, 158]. In MoCap, where pose frequencies are highly skewed, this manifests as strong reconstructions for commonplace poses and weaker ones for rare configurations. However, reconstruction error is not a reliable rarity detector in general. Prior work has shown that atypical examples can still lie close to a learned manifold and be reconstructed well [159]. Our findings on human pose corroborate this: rare poses may show low reconstruction error, whereas frequent poses sometimes yield higher error, undermining reconstruction loss as a proxy for rarity (see Figure 4.2 vs. Figure 4.3).

Motivated by advances in out-of-distribution detection, we therefore adopt a Mahalanobis-distance-based criterion. As noted by [159], it is scale invariant and explicitly accounts for inter-dimensional correlations—both crucial for pose data where coordinates are not independent. Integrating this measure (Eq. (4.2)) enables us to flag structurally rare/novel poses that reconstruction error might miss (Figure 4.3):

$$MD(\mathbf{z}) = \sqrt{(\mathbf{z} - \boldsymbol{\mu}_{\boldsymbol{z}})^{\top} \Sigma_z^{-1} (\mathbf{z} - \boldsymbol{\mu}_{\boldsymbol{z}})}, \tag{4.2}$$

where $\mathbf{z}$ is the latent code for a sample, $\boldsymbol{\mu}_{\boldsymbol{z}}$ and Σ_z are the dataset mean and covariance of latent codes, respectively. This measures standardized deviation while respecting the covariance structure.

For completeness, we also define the normalized reconstruction error (Eq. (4.3)), where $\bar{(\cdot)}$ denotes unit normalization by the bounding-box diagonal of input joints

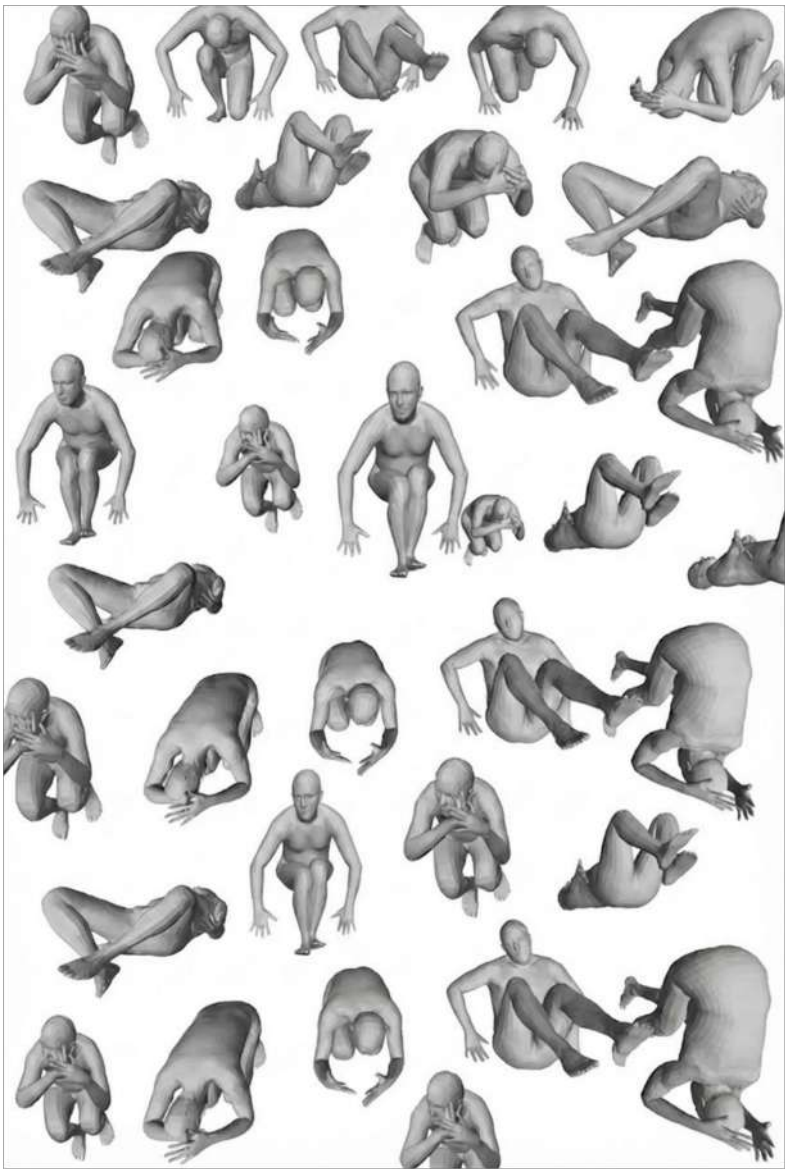


Figure 4.2 Reconstructions of samples exhibiting low RMSE values. Under a conventional reconstruction-based relevance formulation, these samples would be assigned lower weights, as their deviations from the ground truth are minimal. However, when the Mahalanobis distance is employed instead, the same samples yield higher relevance scores, reflecting their true statistical distinctiveness. This formulation allows for more appropriate weighting during the data rebalancing process, which is essential for the success of our approach. Our experiments demonstrate that the Mahalanobis distance more effectively captures subtle structural variations within the dataset that are often overlooked by standard RMSE-based metrics.

and r_{RMSE} is the RMSE between original and reconstructed joints:

$$r_{RMSE} = \sqrt{\frac{1}{J}\sum_{j=1}^{J}\left\|\bar{\ell}^{j} - \bar{\ell}^{j\ddagger}\right\|_2^2} \tag{4.3}$$

with J defining the landmark count.

We consider two relevance kernels $\rho(\epsilon)$:

$$\sigma(\epsilon) = 1 + 2\left(\frac{e^x}{e^x+1} - 0.5\right), \quad x = \frac{\epsilon}{\phi}, \tag{4.4}$$

$$e(\epsilon) = e^{\frac{\epsilon}{\phi}}, \tag{4.5}$$

where ϵ is either the Mahalanobis distance (Eq. (4.2)) or the normalized reconstruction error (Eq. (4.3)), and ϕ controls the scaling. To maintain stable magnitudes, we clamp the exponential-based relevance at $\rho(\epsilon) = 3$, resulting in an effective range $[1, 3]$, whereas the sigmoid-based relevance spans $[1, 2]$. Relevance scaling modulates each sample's contribution to the batch loss, counteracting bias from mean-like poses and ensuring tail examples exert sufficient influence per iteration. Figure 4.4 visualizes exponential weighting; Figure 4.5 shows the sigmoid alternative (here with reconstruction-based ϵ), illustrating when reconstruction can still act as a useful rarity proxy.

Balance via Synthesis. Generative, disentangling models learn low-dimensional manifolds that organize factors of variation. In MoCap, this yields latent spaces where similar poses reside nearby. Even when tail instances are not reconstructed perfectly, latent neighborhoods remain meaningful, enabling interpolation to traverse underrepresented regions and synthesize plausible poses [157, 160]. This is particularly helpful for mitigating bias and redundancy by enriching the tail. We identify tail poses $\boldsymbol{\theta}^{\dagger}$ via thresholding (i.e., $\tau_{threshold}$) of the relevance score in Eq. (4.4). The corresponding anchor codes are

$$\mathcal{A} = \left\{\mathcal{E}(\boldsymbol{\theta}^{\dagger}) \;\middle|\; \rho(\epsilon) > \tau_{threshold}\right\}. \tag{4.6}$$

Our general sampling operator is:

$$\mathcal{S}_{i,j,k,l}(\cdot) = \begin{cases} \varsigma(\mathcal{N}(\mathbf{a}_i, \sigma_{std}),\, \mathcal{N}(\mathbf{a}_j, \sigma_{std}),\, b), & \text{if } \varsigma \in \{LERP, SLERP\} \\ \varsigma(\mathcal{N}(\mathbf{a}_i, \sigma_{std}),\, \mathcal{N}(\mathbf{a}_j, \sigma_{std}),\, \mathcal{N}(\mathbf{a}_k, \sigma_{std}),\, \mathcal{N}(\mathbf{a}_l, \sigma_{std}),\, b), & \text{if } \varsigma \in \{SQUAD, BEZIER, BSPLINE\} \end{cases} \tag{4.7}$$

with $\mathbf{a}_{i,j,k,l}$ randomly selected from $\mathcal{A}$, σ_{std} the standard deviation, b the blending weights, and ς the interpolation function

We evaluate Euclidean and spherical interpolations with varying numbers of control points and continuity—ranging from linear to cubic schemes.

Linear interpolation (LERP).

$$LERP(\mathbf{a}_i, \mathbf{a}_j, b) = (1 - b)\,\mathbf{a}_i + b\,\mathbf{a}_j \tag{4.8}$$

Spherical linear interpolation (SLERP) [161]:

$$SLERP(\mathbf{a}_i, \mathbf{a}_j, b) = \frac{\sin((1-b)\psi)}{\sin(\psi)}\mathbf{a}_i + \frac{\sin(b\psi)}{\sin(\psi)}\mathbf{a}_j \tag{4.9}$$

where ψ is the angular distance between the two latent codes [148].

Spherical cubic spline (SQUAD).

$$\begin{aligned} SQUAD(\mathbf{a}_i, \mathbf{a}_j, \mathbf{a}_k, \mathbf{a}_l, b) = SLERP(&SLERP(\mathbf{a}_i, \mathbf{a}_j, b),\\ &SLERP(\mathbf{a}_k, \mathbf{a}_l, b),\\ &2b(1-b)) \end{aligned} \tag{4.10}$$

Bezier curve (iterated LERP over four control points):

$$BEZIER(\mathbf{a}_i, \mathbf{a}_j, \mathbf{a}_k, \mathbf{a}_l, b) = (1-b)^3\mathbf{a}_i + 3(1-b)^2 b\,\mathbf{a}_j + 3(1-b)b^2\,\mathbf{a}_k + b^3\mathbf{a}_l \tag{4.11}$$

B-spline (piecewise cubic with knot sequence):

$$B\text{-}SPLINE(\mathbf{a}_i, \mathbf{a}_j, \mathbf{a}_k, \mathbf{a}_l, b) = B_{i,3}(b)\mathbf{a}_i + B_{j,3}(b)\mathbf{a}_j + B_{k,3}(b)\mathbf{a}_k + B_{l,3}(b)\mathbf{a}_l \tag{4.12}$$

with $B_{\cdot,3}(\cdot)$ the cubic B-spline basis functions.

Practically, we draw samples from Gaussians centered at two distinct random anchors $i \neq j$, with standard deviation $\sigma_{\mathbf{std}}$, and blend them via SLERP [161] using $b \sim \mathcal{U}(0, 1)$. For higher-order schemes, we sample around four anchors to obtain the additional control points. High-order interpolation helps avoid dead regions in the manifold—since not all directions yield meaningful syntheses [162, 163]—and improves plausibility [164], as shown in Figure 4.6, while increasing diversity through additional controls.

Even if tail poses are not perfectly reconstructed, modern generative models still induce coherent latent neighborhoods, enabling traversal that yields realistic intermediate poses. We, thus, couple anchor selection (via relevance) with interpolation-based synthesis to enrich tail coverage.

For completeness, we also present the two-anchor sampling operator used in ablations:

$$\mathcal{S}_{i,j}(\cdot) = \varsigma(\mathcal{N}(\mathbf{a}_i, \sigma_{std}),\, \mathcal{N}(\mathbf{a}_j, \sigma_{std}),\, b), \quad \mathbf{a}_{i,j} \in_R \mathcal{A}, \tag{4.13}$$

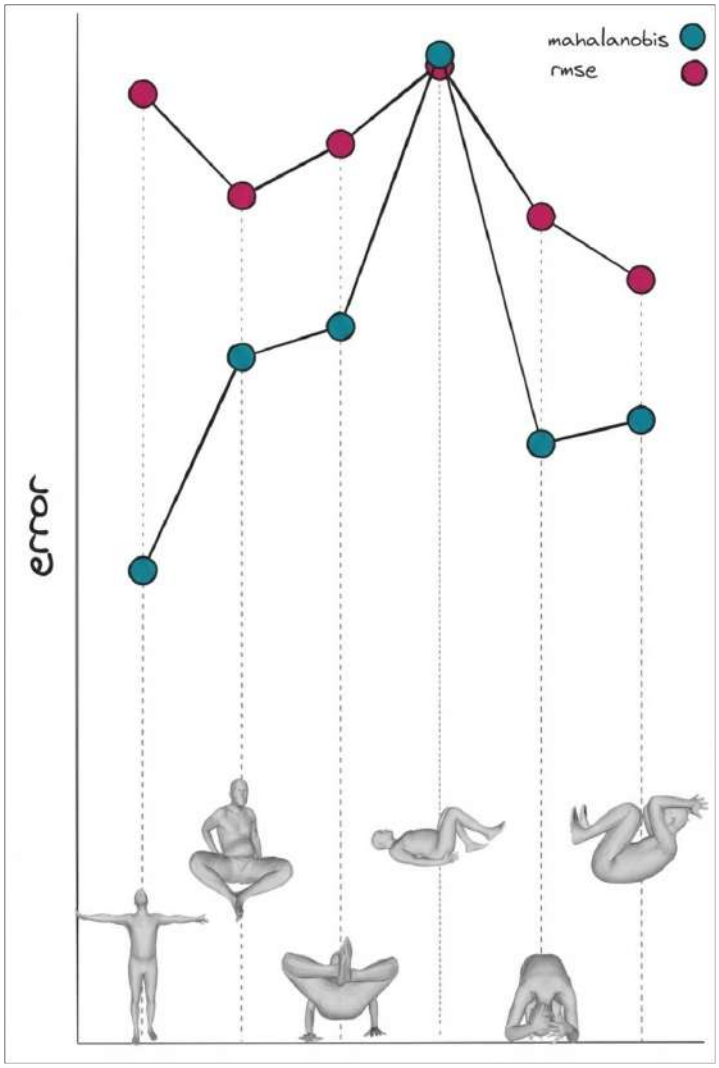


Figure 4.3 Rare poses may yield reconstruction errors of comparable magnitude to those of frequent poses (e.g., standing), which makes simple reconstruction-based metrics ineffective for identifying underrepresented samples and therefore inadequate for proper tail reweighting. This limitation arises because reconstruction error tends to correlate more strongly with pose complexity than with pose frequency in the training distribution. In contrast, the Mahalanobis distance produces distinct error magnitudes across such cases, providing a more reliable indicator of sample rarity and improving the overall effectiveness of the rebalancing strategy. By accounting for the covariance structure of the learned feature space, it captures genuine distributional distance rather than surface-level reconstruction difficulty.

with $b \sim \mathcal{U}(0, 1)$ as above.

Nonlinear interpolation mitigates dead directions [162, 163] and increases sample plausibility [164]; see Figure 4.6.

4.2.3 Landmark Estimation

To showcase the benefits of our balancing strategy, we design a model that unifies denoising, solving, and hallucination—going beyond pipelines that focus solely on labeling [150, 151] or solving [153, 154]. In contrast to approaches that regress directly from raw marker coordinates [151, 153, 154], we adopt a CNN operating on structured heatmaps [12, 152]. This choice improves convergence and accuracy and enables robust multi-view fusion in a single network, akin to [12].

Concretely, augmented and corrupted input markers $\tilde{\boldsymbol{\ell}}^{\star}_{in}$ are normalized and rasterized into depth-like maps. Unlike [12], which learns separate front/back depth, we employ marginal heatmap regression [165, 166] with known gravity direction along y axis. We render orthographic xy and yz views that share the y-axis and fuse their predictions. Center-of-mass regression [167–170] yields normalized landmark positions $\tilde{\boldsymbol{\ell}}_{est}$, using the average expectation for y as in [165, 166]. Because our inputs are already 3D, we avoid additional viewpoint classification needed for monocular imagery, solving a single 2D heatmap task to recover normalized 3D coordinates.

Supervision combines a Jensen–Shannon divergence on heatmaps with a robust coordinate loss:

$$\mathcal{L}_{JS}(\mathbf{H}_{gt}, \mathbf{H}_{est}) = \frac{1}{2} D_{KL}(\mathbf{H}_{gt}, M) + \frac{1}{2} D_{KL}(\mathbf{H}_{est}, M), \tag{4.14}$$

where D_{KL} is the Kullback–Leibler divergence, $M = \frac{1}{2}(\mathbf{H}_{gt} + \mathbf{H}_{est})$, and $\mathbf{H}_{gt}, \mathbf{H}_{est}$ the ground-truth and estimated heatmaps accordingly.

The Welsch penalty on normalized coordinates

$$\mathcal{L}^{\nu}_{w}(\tilde{\boldsymbol{\ell}}_{gt}, \tilde{\boldsymbol{\ell}}_{est}) = 1 - \exp\Big(- \frac{\|\tilde{\boldsymbol{\ell}}_{gt} - \tilde{\boldsymbol{\ell}}_{est}\|^2}{2\nu^2} \Big), \tag{4.15}$$

with $\nu > 0$ set to 0.05. Here, $\mathcal{L}_{JS}$ improves heatmap alignment [171], while $\mathcal{L}^{\nu}_{w}$ [172, 173] stabilizes coordinate regression and supports sub-pixel accuracy.

Figure 4.1 summarizes the training loop. Each mini-batch mixes synthetic samples from Eqs. (4.1) and (4.13). Sample contributions are scaled by the relevance functions from Section 4.2.2.

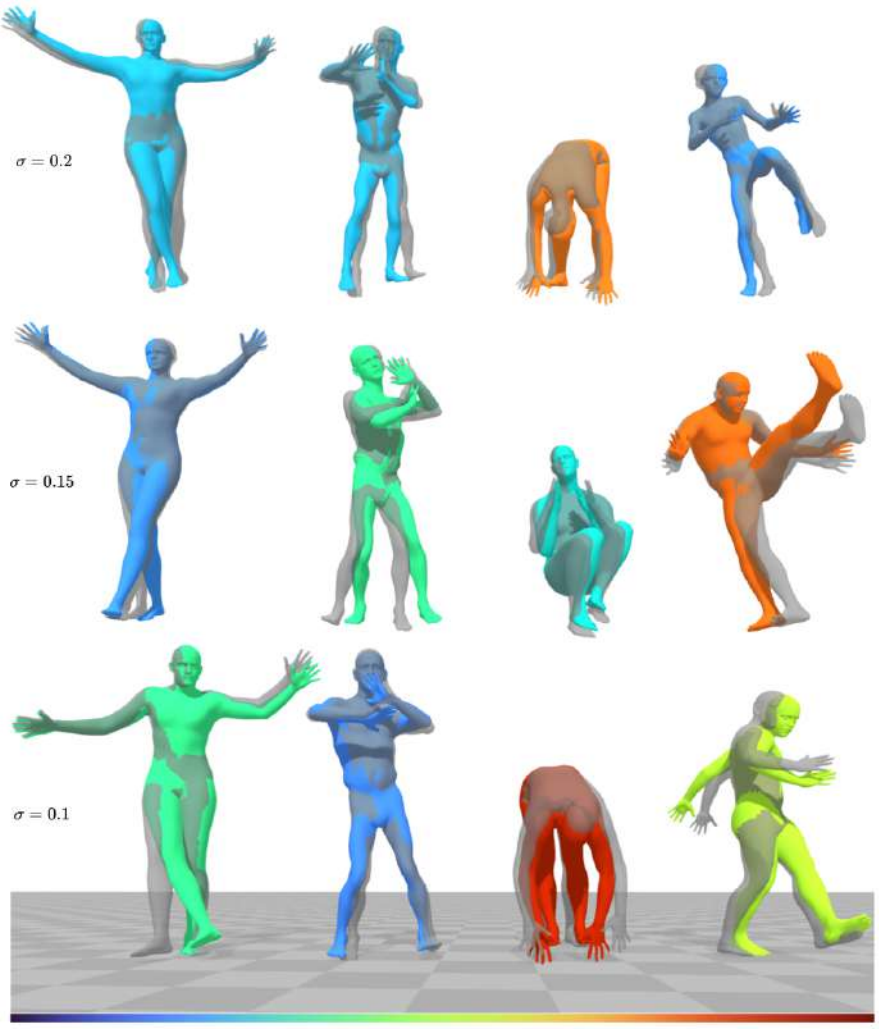


Figure 4.4 Color-coded visualization of autoencoded poses. Each pose is represented according to its relevance weight ρ and corresponding uncertainty σ, computed using the exponential-based relevance function and the reconstruction error metric. It is evident that more difficult poses exhibit bigger weights.

4.2.4 VAE

Beyond VAEs, alternative generative priors for pose include GAN-based models [174] and methods that embed poses onto structured manifolds [175]. We select an autoencoding generative prior because it simultaneously (i) reconstructs poses, (ii) generates rare cases, and (iii) provides latent statistics used by our relevance mechanism. While VAEs can be mean-centered and may contain "dead" latent regions, careful sampling and relevance-based reweighting allow us to exploit them effectively for tail synthesis and balanced training.

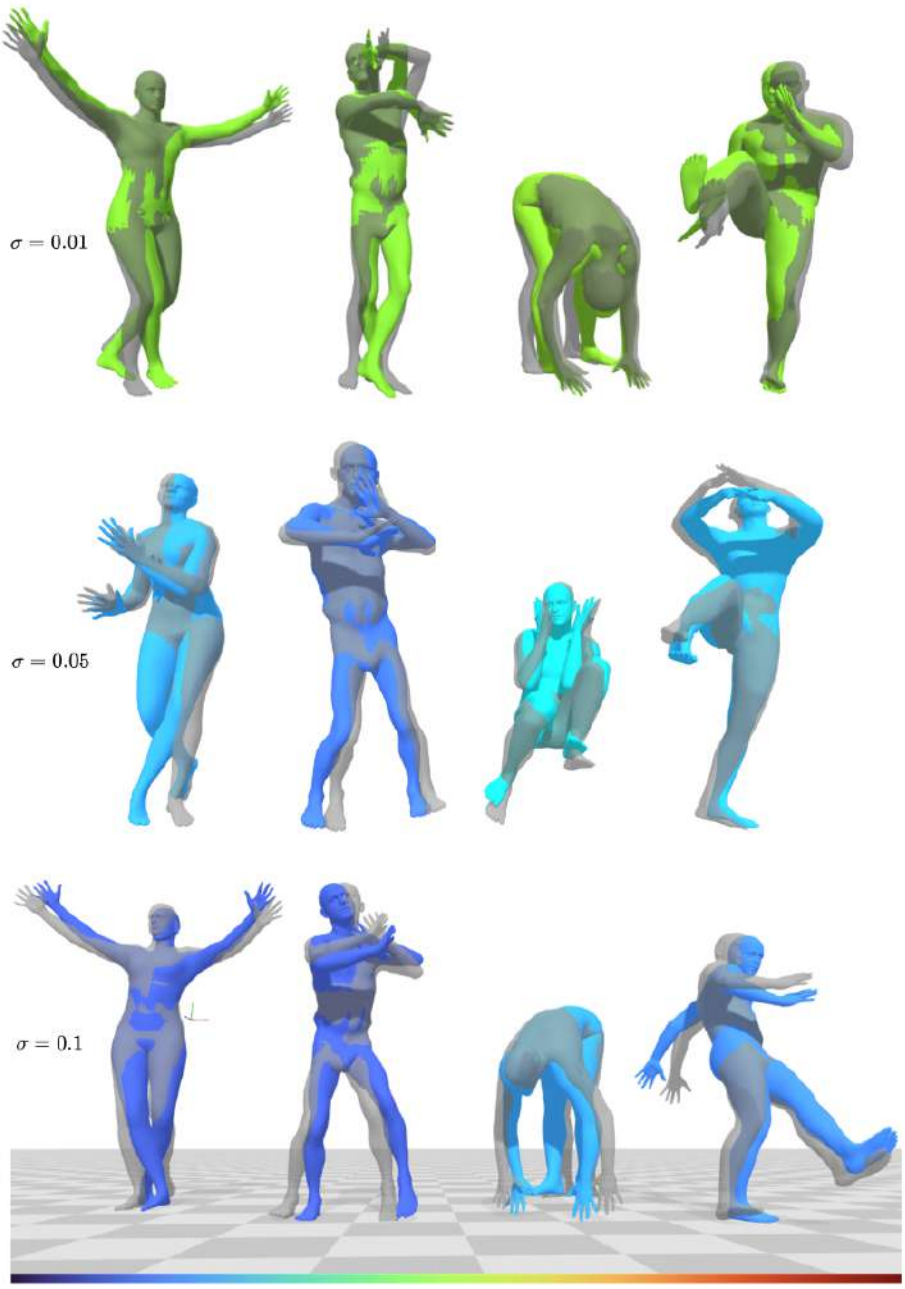


Figure 4.5 Color-coded visualization of autoencoded poses. Each pose is represented according to its relevance weight ρ and corresponding uncertainty σ, computed using the sigmoid-based relevance function and the reconstruction error metric.

We introduce **RVPoser**, architecturally similar to VPoser [4] but with three small modifications: (a) removal of batch normalization [176] before the encoder's first fully-connected layer; (b) no dropout in the decoder; and (c) no activation after

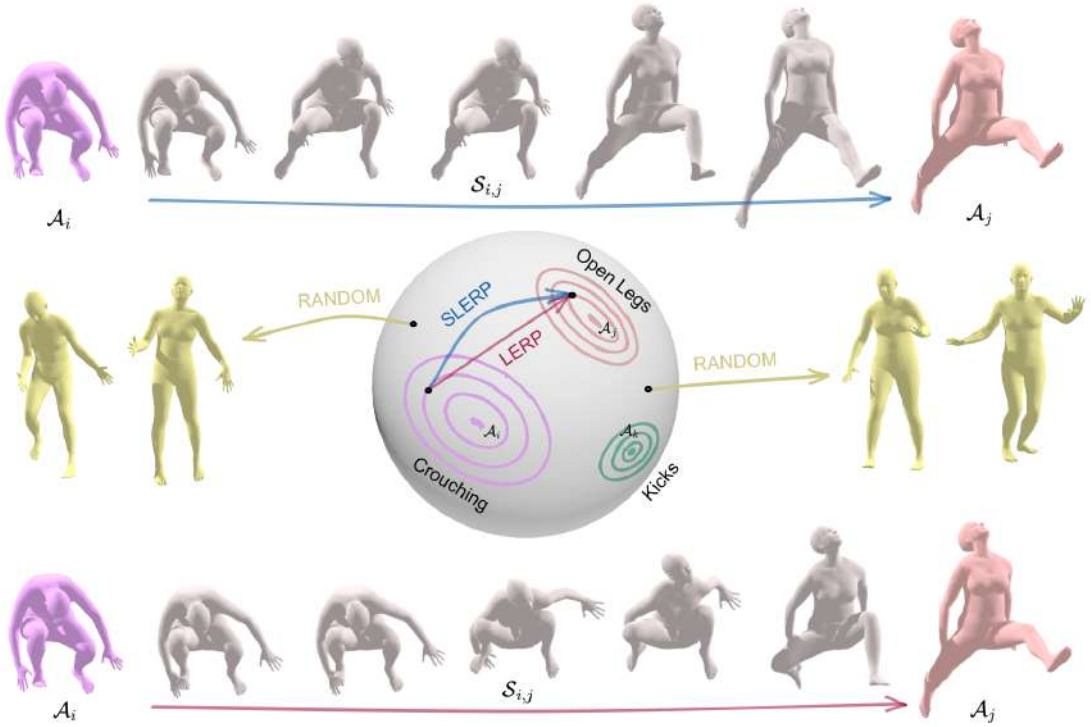


Figure 4.6 Rare-pose oversampling using latent anchors $\mathcal{A}$. Random blending of latent vectors through **non-linear** interpolation methods (e.g., SLERP, SQUAD, Bezier, B-spline) produces more diverse and realistic tail samples. In contrast, **linear** interpolation yields less diversity and may generate implausible poses, while **random** sampling introduces distributional bias.

the decoder's final fully-connected layer. Despite their simplicity, these choices yield performance gains in our setting. The VAE is trained with

$$\mathcal{L}_{VAE} = \lambda_1 \mathcal{L}_{KL} + \lambda_2 \mathcal{L}_{rec} + \lambda_3 \mathcal{L}_{orth}, \tag{4.16}$$

$$\mathcal{L}_{KL} = \Psi\big(D_{KL}(q_\theta(\mathbf{z}) \,\|\, \mathcal{N}(0, I))\big), \tag{4.17}$$

$$\mathcal{L}_{rec} = \|\mathbf{v} - \hat{\mathbf{v}}\|_2^2, \tag{4.18}$$

$$\mathcal{L}_{orth} = \frac{\mathrm{Tr}(\mathbf{R}^\top \hat{\mathbf{R}}) - 1}{2}, \tag{4.19}$$

where $\mathbf{z} \in \mathbb{R}^{32}$ is the latent code; $\mathbf{R} \in \mathbb{SO}(3)^P$ are pose rotations; $\hat{\mathbf{R}}$ are decoded rotations; and $\mathbf{v}, \hat{\mathbf{v}}$ denote ground-truth and predicted vertices (covering angular and 3D joint-position errors). To regularize the KL term and mitigate posterior collapse, we apply the Charbonnier penalty $\Psi(\mathbf{x}) = \sqrt{1 + \mathbf{x}^2} - 1$ [177]. Thus, Eqs. (4.17) and (4.18) enact the conventional VAE trade-off between reconstruction and Gaussianized latent structure, while Eqs. (4.18) and (4.19) encourage a valid rotational

latent space. Training uses AdamW [178] to restrain weight growth and reduce overfitting.

4.3 Experiments

4.3.1 Training and Evaluation Datasets

Training: We train on a broad collection of MoCap datasets unified in AMASS [2], using their body-model parameterizations. Specifically, we include the CMU corpus (large, diverse motions such as walking, running, dancing, etc.), Transitions (activity changes like sit/stand and pick/carry), PosePrior [179] (statistical pose coverage), HumanEva [180] (multiple subjects and activities), and ACCAD [181] (rich actions spanning dance, martial arts, and sports). We further incorporate TotalCapture [182] (5 subjects, 37 action classes), DFaust [183] (10 subjects, 129 motion types), and a CNRS subset (2 subjects, 79 motions).

Validation: For validation we use THuman 2.0 [184] (5 subjects, extreme poses) and a manually curated collection of rare configurations (the "Tail" set) comprising 274 challenging instances grouped coarsely as *crossed legs*, *crossed arms*, *kicks*, and *crouching*. This set is used to stress-test long-tail regression and we intend to release it publicly to support further research. We additionally evaluate on GeneBody [185], which contains demanding actions such as yoga and dance.

Preprocessing: Following recent practice [151, 153, 154], we adopt a preprocessing pipeline that first augments and then corrupts training inputs. Corruption exploits the synthetic nature of parametric markers to approximate real capture conditions (ghosts, occlusions, measurement noise), while augmentations broaden variability.

Augmentations: To model body-shape variability, we apply a two-stage perturbation of shape coefficients. First, we shift all coefficients by a uniform offset $u \sim \mathcal{U}(-1, 1)$:

$$\boldsymbol{\beta}' = \boldsymbol{\beta} + u\,\mathbf{1}, \tag{4.20}$$

where $\mathbf{1}$ denotes the all-ones vector. Then, we randomly replace a small subset of components with draws from a standard normal. Let $\mathcal{I} \subset \{1, \ldots, 10\}$ be a randomly sampled index set; in our runs $|\mathcal{I}| \leq 3$ with $\mathcal{I} \subseteq \{1, 2, 3\}$. The final augmented shape vector is:

$$\boldsymbol{\beta}''_i = \begin{cases} \boldsymbol{\beta}'_i, & \text{if } i \notin \mathcal{I} \\ \mathcal{N}(0, 1), & \text{if } i \in \mathcal{I} \end{cases} \tag{4.21}$$

This two-stage scheme shifts the overall shape distribution while occasionally randomising the dominant principal components, broadening the diversity of body shapes seen during training.

Corruption. We simulate realistic marker corruption through three mechanisms: *occlusions*, *ghost markers*, and *measurement noise*.

Let

$$\mathbf{p} = (\mathbf{p_1}, \ldots, \mathbf{p_n}), \qquad \mathbf{p_i} \in \mathbb{R}^3 \tag{4.22}$$

denote the original set of n marker positions.

Occlusion Simulation. To emulate occlusions, we randomly remove a subset of markers. First, we sample the number of removed markers:

$$k \sim \mathcal{U}\{m, n'\}, \qquad m \leq k \leq n' \leq n, \tag{4.23}$$

where m and n' define the minimum and maximum number of occluded markers. Next, we uniformly sample an index set:

$$\mathcal{I} \subset \{1, \ldots, n\}, \qquad |\mathcal{I}| = k, \tag{4.24}$$

without replacement, and remove all markers $\{\mathbf{p}_i \mid i \in \mathcal{I}\}$ from $\mathbf{p}$.

Ghost Marker Synthesis. To simulate ghost detections, we generate artificial markers from a Gaussian distribution fitted to the remaining visible markers, following [151]. Let $\boldsymbol{\mu_{marker}} \in \mathbb{R}^3$ denote the per-axis median vector and $\mathbf{\Sigma_{marker}} \in \mathbb{R}^{3\times3}$ the sample covariance matrix computed from the observed markers. We draw n_{ghost} synthetic markers

$$g_j \sim \mathcal{N}(\boldsymbol{\mu_{marker}}, \mathbf{\Sigma_{marker}}), \qquad j = 1, \ldots, n_{\text{ghost}}, \tag{4.25}$$

and append them to the marker set.

Measurement Noise. To model sensor noise, we randomly select a subset of N markers uniformly without replacement. For each selected marker $\mathbf{p}_i$, we add an independent bounded offset

$$\mathbf{o_i} \sim \mathcal{U}(-M_{\text{noise}}, M_{\text{noise}})^3, \tag{4.26}$$

and define the perturbed marker

$$\mathbf{p}'_i = \mathbf{p}_i + \mathbf{o}_i. \tag{4.27}$$

Markers not selected remain unchanged.

4.3.2 Evaluation Metrics

MoCap Metrics: We report standard errors—RMSE and MPJE—consistent with prior work. RMSE is

$$RMSE = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\frac{1}{J}\sum_{j=1}^{J}\left\|\boldsymbol{\ell}_{gt}^{(i,j)} - \boldsymbol{\ell}_{est}^{(i,j)}\right\|_2}, \tag{4.28}$$

where N is the number of samples and J the landmark count. We further use a PCK-style accuracy with thresholds $\tau \in \{10, 30, 70\}$ mm (PCK1, PCK3, PCK7), and also report limb-specific scores for arms and legs:

$$PCK = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{J}\sum_{j=1}^{J}\left[\left\|\boldsymbol{\ell}_{gt}^{(i,j)} - \boldsymbol{\ell}_{est}^{(i,j)}\right\|_2 < \tau\right]. \tag{4.29}$$

Generative Metrics: Following [186], we evaluate (i) realism via FID and (ii) diversity via average pairwise distance. For FID, we extract features from 1,052 generated and real poses, and compute

$$FID = \|\mu_{real} - \mu_{gen}\|_2^2 + \mathrm{Tr}\left(\boldsymbol{\Sigma}_{real} + \boldsymbol{\Sigma}_{gen} - 2\sqrt{\boldsymbol{\Sigma}_{real}\boldsymbol{\Sigma}_{gen}}\right), \tag{4.30}$$

with $(\mu, \boldsymbol{\Sigma})$ being the feature means and covariances. For diversity, we generate and re-encode 1,052 samples, split into two groups of 526, and report

$$DIV = \frac{1}{N_{div}}\sum_{i=1}^{N_{div}}\|\mathbf{z}_i - \tilde{\mathbf{z}}_i\|_2, \quad N_{div} = 526, \tag{4.31}$$

where $\mathbf{z}$ and $\tilde{\mathbf{z}}$ are re-encoded vectors from each subset.

4.3.3 Analysis

Across three heterogeneous benchmarks, our sampling+reweighting strategy consistently improves performance. On THuman (Table 4.1)—which blends routine and challenging poses—our method preserves accuracy on standard configurations while substantially boosting scores on the Tail set (Table 4.2) across most metrics.

Each component (sampling or reweighting) helps on its own, and the combination is strongest. Results on GeneBody (complex yoga) further corroborate these gains (Table 4.3)

Figure 4.10 provides qualitative evidence: high-order interpolation noticeably elevates pose plausibility and reduces error, while reweighting improves a different facet

		RMSE ↓	MPJPE ↓	PCK1 ↑			PCK3 ↑			PCK7 ↑		
				all	arms	legs	all	arms	legs	all	arms	legs
	[153]	22.01 *mm*	17.97 *mm*	27.15%	19.01%	26.10%	91.45%	87.43%	95.05%	98.01%	97.13%	98.01%
	[12]	21.81 *mm*	17.68 *mm*	28.10%	19.89%	26.55%	91.89%	87.12%	95.11%	98.55%	97.10%	98.49%
	Ours	21.41 *mm*	17.57 *mm*	28.69%	21.21%	27.33%	92.08%	87.50%	95.01%	98.59%	97.30%	98.89%
sampling	BMSE	22.21 *mm*	18.00 *mm*	25.51%	19.55%	25.02%	91.90%	86.10%	95.05%	98.62%	97.05%	98.15%
sampling	RANDOM	21.52 *mm*	16.78 *mm*	31.60%	24.11%	30.11%	92.49%	88.10%	95.11%	98.11%	97.10%	98.95%
sampling	LERP	21.59 *mm*	16.80 *mm*	29.48%	21.11%	29.07%	92.68%	88.25%	95.15%	98.58%	97.89%	99.01%
sampling	SLERP	20.43 *mm*	16.29 *mm*	30.41%	22.62%	29.02%	93.67%	88.40%	95.32%	98.92%	98.04%	99.05%
sampling	SQUAD	**18.80** *mm*	**15.33** *mm*	32.94%	25.03%	31.20%	**94.81%**	89.10%	95.41%	**99.19%**	**98.20%**	**99.31%**
sampling	BEZIER	18.94 *mm*	15.51 *mm*	32.90%	**25.03%**	31.23%	94.56%	**89.92%**	**96.15%**	99.04%	98.16%	99.22%
sampling	B-spline	20.20 *mm*	16.06 *mm*	**33.88%**	24.78%	**36.31%**	93.49%	88.47%	94.99%	98.82%	98.07%	98.81%
relevance	e(R)	20.65 *mm*	16.99 *mm*	31.11%	22.1%	28.15%	93.01%	88.5%	95.34%	98.55%	97.45%	98.89%
relevance	σ(R)	20.60 *mm*	16.67 *mm*	30.99%	22.4%	28.32%	92.79%	**88.65%**	**95.5%**	98.61%	97.65%	98.91%
relevance	e(D)	20.83 *mm*	16.59 *mm*	31.5%	**25.54%**	29.18%	**93.39%**	88.56%	94.75%	98.69%	**97.72%**	98.76%
relevance	σ(D)	**20.49** *mm*	**16.51** *mm*	**31.57%**	23.26%	**29.35%**	92.72%	86.63%	94.84%	**98.72%**	97.62%	**98.95%**
ortho	e(D) + SQUAD	20.57 *mm*	16.48 *mm*	31.07%	23.54%	29.02%	93.44%	88.34%	94.84%	98.62%	97.59%	98.72%
ortho	σ(D)+ SQUAD	**18.69** *mm*	**15.20** *mm*	**35.59%**	**28.52%**	**36.43%**	94.53%	**89.99%**	96.06%	99.09%	**98.32%**	99.22%

Table 4.1: **Results on the THuman dataset.** High-order interpolation methods outperform conventional interpolation approaches as well as the state-of-the-art balancing method BMSE [11]. Similarly, the Mahalanobis-based reweighting scheme proves more effective than the reconstruction-based alternative across most evaluation metrics. The final rows highlight the synergistic effect of combining relevance weighting and oversampling, leading to consistent performance improvements over the baseline model.

		RMSE ↓	MPJPE ↓	PCK1 ↑			PCK3 ↑			PCK7 ↑		
				all	arms	legs	all	arms	legs	all	arms	legs
	[153]	36.10 *mm*	32.60 *mm*	20.10%	12.13%	23.23%	80.23%	77.10%	80.33%	93.12%	93.43%	94.92%
	[12]	36.05 *mm*	32.11 *mm*	21.01%	13.41%	24.53%	80.12%	77.43%	80.73%	93.23%	93.1%	95.12%
	Ours	35.80 *mm*	31.15 *mm*	22.04%	14.99%	25.68%	80.27%	77.94%	81.00%	94.31%	93.%	95.21%
sampling	BMSE	32.90 *mm*	**25.21** *mm*	**27.66%**	16.21%	26.13%	81.98%	77.32%	81.10%	94.92%	93.10%	95.10%
sampling	RANDOM	35.80 *mm*	31.84 *mm*	23.00%	14.56%	26.01%	81.81%	77.46%	80.20%	95.70%	93.01%	95.30%
sampling	LERP	33.50 *mm*	28.19 *mm*	25.02%	12.30%	23.22%	79.82%	76.31%	81.10%	95.22%	93.05%	95.10%
sampling	SLERP	33.25 *mm*	25.82 *mm*	24.83%	14.23%	30.29%	80.85%	77.08%	82.16%	95.48%	93.18%	**95.75%**
sampling	SQUAD	**32.68** *mm*	25.53 *mm*	25.11%	**16.97%**	**30.35%**	**83.78%**	**79.78%**	**84.15%**	95.55%	93.72%	94.63%
sampling	BEZIER	34.01 *mm*	26.26 *mm*	24.38%	15.37%	26.14%	82.10%	78.07%	81.34%	**95.81%**	93.69%	95.66%
sampling	B-spline	34.82 *mm*	27.49 *mm*	23.30%	14.53%	24.81%	80.25%	74.84%	81.83%	95.18%	**94.02%**	95.23%
relevance	e(R)	34.2 *mm*	26.1 *mm*	23.10%	12.01%	25.10%	80.88%	74.10%	81.02%	94.95%	93.12%	95.01%
relevance	σ(R)	33.9 *mm*	26.4 *mm*	23.61%	12.11%	26.55%	81.00%	74.10%	80.10%	95.21%	93.02%	94.33%
relevance	e(D)	33.81 *mm*	25.82 *mm*	**25.53%**	**18.32%**	**28.67%**	**81.11%**	**75.45%**	**82.84%**	94.91%	**93.62%**	94.93%
relevance	σ(D)	**33.79** *mm*	**25.7** *mm*	22.63%	13.18%	27.27%	79.74%	74.52%	80.40%	**95.21%**	92.60%	**95.45%**
ortho	e(D) + SQUAD	34.06 *mm*	26.59 *mm*	25.21%	14.28%	30.16%	82.78%	78.21%	83.33%	95.80%	**95.94%**	95.35%
ortho	σ(D)+ SQUAD	**32.54** *mm*	**24.48** *mm*	**28.15%**	**18.90%**	29.68%	**83.89%**	**79.85%**	82.90%	95.54%	95.60%	94.33%

Table 4.2: **Results on the TAIL dataset.** Among the tested interpolation schemes, **SQUAD** provides the most effective strategy for sampling the latent space and improving performance under data imbalance. Interestingly, BMSE [11] also yields competitive results, ranking second across most metrics. Nevertheless, the Mahalanobis-based reweighting scheme consistently outperforms the reconstruction-based alternative, while the combined use of sampling and relevance weighting further enhances performance—except for the PCK@7 metric.

of the error landscape. Together, they yield the best overall outcome. We additionally compare to the balancing method of [11]. Although it enhances Tail performance (Table 4.2), it degrades on more evenly distributed data such as THuman (Table 4.1). Our approach improves rare-pose handling without sacrificing performance on balanced sets, and using a single VAE both for sampling and relevance outperforms alternatives in human-pose regression.

Which metric best identifies rare human poses? Mahalanobis distance (Figure 4.7) consistently outperforms reconstruction-based metrics (RMSE) for reweighting across datasets. Intuitively, latent codes for in-distribution poses follow Gaussian structure, so standardized deviations that respect covariance are more discriminative for rarity than raw reconstruction error. Figure 4.3 shows RMSE conflates common and rare cases (e.g., standing vs. atypical), whereas Mahalanobis separates them and yields more effective weighting.

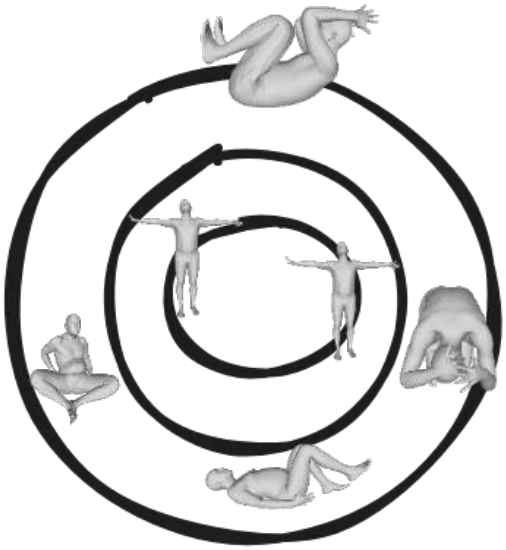


Figure 4.7 Mahalanobis distance in human pose data. Each concentric circle represents the density and distribution of human poses within the learned VAE latent space. The Mahalanobis distance measures the relative displacement of poses from the mean pose (center), accounting for inter-pose covariance. This metric proves substantially more effective than reconstruction-based errors for identifying and reweighting rare poses, thereby improving dataset balance.

Optimal sampling method for data generation: High-order interpolation dominates LERP/SLERP in Tables 4.1 to 4.3. SQUAD scores best overall, followed

by B-spline and Bezier. Notably, even Euclidean high-order curves (Bezier/B-spline) outperform SLERP, underscoring the value of additional control points for diversity. Broader coverage around anchors materially improves Tail performance without harming results on balanced datasets (Table 4.1).

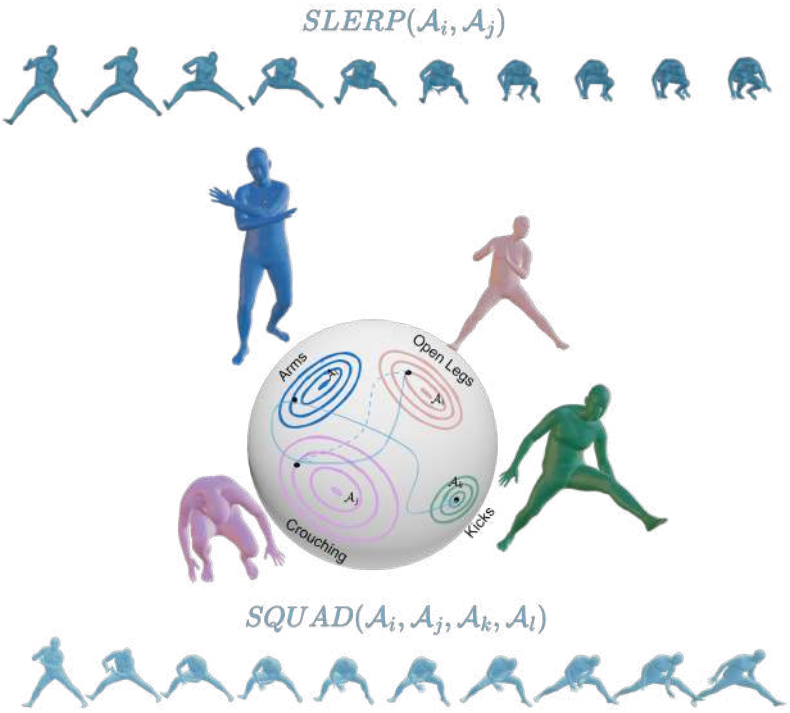


Figure 4.8 Comparison of SLERP and SQUAD interpolation. While SLERP interpolates between two latent $\mathcal{A}$, SQUAD enables interpolation among four, allowing the generation of more diverse and realistic samples within the latent space. This increased expressivity is crucial for effective oversampling of rare poses. Similar benefits are observed with other high-order interpolation methods, such as Bezier and B-spline.

Do sampling and reweighting act synergistically? The "ortho" rows in Tables 4.1 to 4.3 combine best-of sampling with best-of reweighting and deliver the strongest results, as echoed by Figure 4.10. On Tail, we observe a slight departure from perfect orthogonality: reweighting tightens tolerance to holistic pose errors (legs in PCK1/3/7), while oversampling edges it overall—aligned with broader evidence in imbalanced regression. These results suggest complementary effects and highlight remaining open questions in balancing continuous targets.

Does the framework transfer across architectures? Our scheme is model-agnostic (Table 4.4). Applying both reweighting and oversampling to [12] improves

		RMSE ↓	MPJPE ↓	PCK1 ↑			PCK3 ↑			PCK7 ↑		
				all	arms	legs	all	arms	legs	all	arms	legs
	[153]	185.10 *mm*	104.22 *mm*	29.15%	26.45%	26.12%	58.56%	68.10%	50.13%	73.22%	79.12%	58.92%
	[12]	190.32 *mm*	105.33 *mm*	28.01%	25.19%	27.13%	59.92%	69.01%	51.12%	73.45%	80.12%	59.81%
	Ours	170.31 *mm*	103.83 *mm*	29.04%	27.06%	28.02%	60.91%	69.11%	51.17%	73.85%	80.39%	59.9%
sampling	BMSE	145.21 *mm*	99.28 *mm*	29.46%	28.02%	27.93%	60.02%	69.78%	51.23%	73.21%	79.89%	60.01%
sampling	RANDOM	150.21 *mm*	102.20 *mm*	27.35%	28.45%	24.29%	60.21%	69.01%	50.23%	75.31%	80.01%	60.21%
sampling	LERP	145.44 *mm*	104.43 *mm*	27.87%	30.00%	25.43%	60.06%	68.75%	50.23%	75.32%	80.32%	59.99%
sampling	SLERP	155.87 *mm*	102.25 *mm*	29.30%	33.89%	26.88%	58.52%	65.28%	50.33%	72.46%	76.50%	59.38%
sampling	SQUAD	**133.22** *mm*	**83.37** *mm*	29.15%	**34.72%**	**28.99%**	**62.56%**	**77.78%**	**54.21%**	**78.17%**	**88.94%**	**65.79%**
sampling	BEZIER	134.71 *mm*	84.42 *mm*	**31.06%**	34.39%	28.96%	62.54%	71.11%	53.96%	77.89%	84.17%	64.54%
sampling	B-spline	165.79 *mm*	101.42 *mm*	28.35%	31.94%	26.75%	62.30%	76.39%	49.54%	78.13%	87.72%	60.58%
relevance	e(R)	110.12 *mm*	101.23 *mm*	28.34%	31.23%	25.65%	58.99%	71.09%	52.35%	75.21%	81.10%	59.92%
relevance	σ(R)	120.12 *mm*	105.32 *mm*	28.12%	31.23%	26.15%	60.01%	70.82%	52.35%	75.42%	81.23%	59.98%
relevance	e(D)	**105.07** *mm*	**71.08** *mm*	**29.59%**	**32.78%**	**27.75%**	**62.04%**	**72.61%**	53.17%	**76.19%**	**82.50%**	**66.33%**
relevance	σ(D)	174.19 *mm*	118.9 *mm*	28.39%	30.00%	26.92%	59.43%	63.83%	**54.33%**	72.37%	69.94%	64.50%
ortho	e(D) + SQUAD	**100.76** *mm*	**61.71** *mm*	29.72%	34.33%	26.38%	**67.69%**	**80.39%**	**56.00%**	**83.15%**	**95.33%**	68.75%
ortho	σ(D)+ SQUAD	108.74 *mm*	65.33 *mm*	**31.52%**	**35.06%**	**29.29%**	62.85%	70.56%	54.96%	81.48%	87.83%	**69.00%**

Table 4.3: **Results on the GeneBody dataset.** This dataset contains complex yoga poses that are severely underrepresented in the training data. In this scenario, the advantages of our proposed approach become even more pronounced, demonstrating its ability to handle extreme pose variations and data imbalance effectively.

benchmarks across datasets, confirming cross-architecture compatibility. Incorporating [11] into [12] helps on Tail/GeneBody but not THuman; in contrast, our approach raises tail performance without degrading balanced-set accuracy.

		RMSE ↓	MPJPE ↓	PCK1 ↑	PCK3 ↑	PCK7 ↑
TH2	[12]	21.81 *mm*	17.68 *mm*	28.10%	91.89%	98.55%
TH2	[12] + BMSE	22.01 *mm*	18.05 *mm*	25.05%	91.79%	98.65%
TH2	[12] + Ours	**19.95** *mm*	**16.45** *mm*	**32.30%**	**92.45%**	**98.70%**
TAIL	[12]	36.05 *mm*	32.11 *mm*	21.01%	80.12%	93.23%
TAIL	[12] + BMSE	33.34 *mm*	26.15 *mm*	**25.35%**	81.03%	93.52%
TAIL	[12] + Ours	**33.01** *mm*	**25.55** *mm*	24.99%	**82.75%**	**93.90%**
GENE	[12]	190.32 *mm*	105.33 *mm*	28.01%	59.92%	73.45%
GENE	[12] + BMSE	150.15 *mm*	101.01 *mm*	28.50%	60.01%	74.65%
GENE	[12] + Ours	**110.01** *mm*	**70.55** *mm*	**30.15%**	**62.75%**	**80.05%**

Table 4.4: Results from [12] using our proposed training approach. The proposed training strategy is model-agnostic and can be seamlessly integrated into different network architectures, demonstrating its general applicability.

Dependence on the prior's quality: Although the training scheme accepts any VAE prior, performance tracks prior quality. Compared to VPoser [4], our RVPoser (Section 4.2.4) yields larger gains (Table 4.6)—both in pose synthesis (realism/diversity) and when used as a fitting prior, improving angular errors and pose

prediction accuracy (except PCK7).

	Synthesis		Fitting			
	FID ↓	DIV ↑	MAE ↓	PCK1 ↑	PCK3 ↑	PCK 7 ↑
VPoser [4]	9.94	12.11	2.68°	28.83%	89.04%	**99.03%**
RVPoser (Ours)	**8.57**	**13.24**	**1.51°**	**53.72%**	**94.57%**	98.15%

Table 4.5: Quantitative comparison between VPoser [4] and our robust variant (RVPoser). Results are reported for both synthesis and fitting tasks on the THuman 2.0 test set, showing that RVPoser achieves improved robustness and accuracy over the original model.

UMAP visualizations (Figure 4.9) indicate that RVPoser spans a broader portion of the ground-truth manifold, supporting better coverage and stronger tail synthesis.

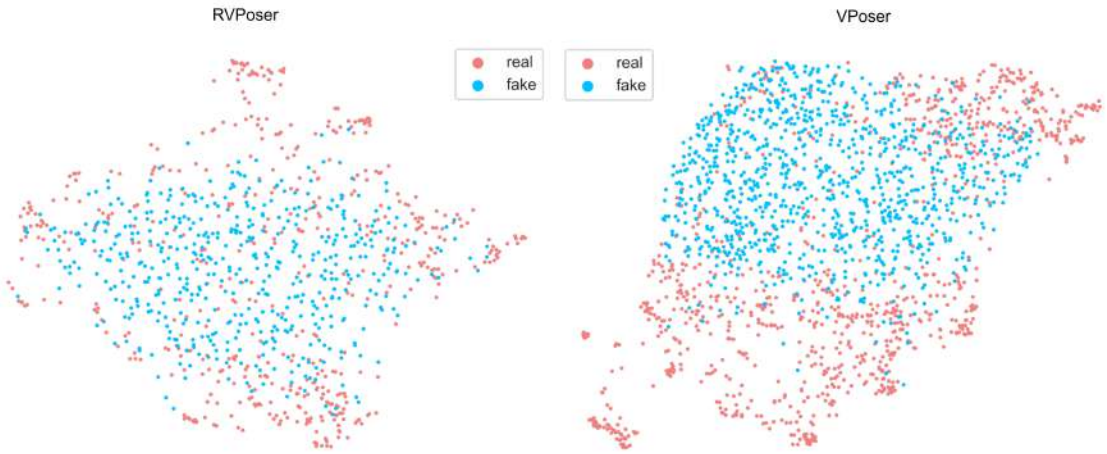


Figure 4.9 UMAP projections [3] of ground-truth ("real") and generated ("fake") samples from RVPoser and VPoser [4]. Compared to VPoser, our RVPoser exhibits broader and denser coverage of the ground-truth pose manifold, enabling the generation of more diverse and realistic samples.

Which architecture is most suitable for MoCap solving? Our U-Net variant surpasses Democap [12] and MoCap-Solver [153] across all benchmarks. Despite modest architectural changes, the cumulative engineering choices materially improve accuracy and, with low latency, are suitable for real-time use.

		RMSE ↓	MPJPE ↓	PCK1 ↑	PCK3 ↑	PCK7 ↑
TH2	Benchmark	21.41 *mm*	17.57 *mm*	28.69%	92.08%	98.59%
	Benchmark + [4]	19.01 *mm*	16.55 *mm*	31.11%	93.33%	98.85%
	Benchmark + Ours	**18.69** *mm*	**15.20** *mm*	**35.59%**	**94.53%**	**99.09%**
TAIL	Benchmark	35.80 *mm*	31.15 *mm*	22.04%	80.27%	94.31%
	Benchmark + [4]	33.45 *mm*	26.01 *mm*	24.35%	82.03%	94.25%
	Benchmark + Ours	**32.54** *mm*	**24.48** *mm*	**25.15%**	**83.29%**	**95.54%**
GENE	Benchmark	170.31 *mm*	103.83 *mm*	29.04%	60.91%	73.85%
	Benchmark + [4]	125.15 *mm*	75.55 *mm*	30.01%	61.01%	75.65%
	Benchmark + Ours	**108.74** *mm*	**65.33** *mm*	**31.52%**	**62.85%**	**81.48%**

Table 4.6: Results of our proposed training framework applied to an alternative VAE [4]. The framework, which combines oversampling and reweighting, consistently improves performance, demonstrating its effectiveness across different latent representations.

Computational complexity of sampling:

Sampling Method	Complexity	Continuity
LERP	$O(1)$	C^0
SLERP	$O(1)$	C^0
SQUAD	$O(1)$	C^1
BEZIER	$O(n)$ for linear, $O(n^2)$ for quadratic, $O(n^3)$ for cubic	Varies, up to C^{n-1}
B-Spline	$O(n)$ to $O(n^2)$ depending on degree and control points	C^{n-1}

Table 4.7: Comparison of computational complexity and continuity across different interpolation methods. Here, n denotes the polynomial degree (for Bézier curves, n = number of control points − 1), while for B-Splines it refers to the number of control points and the resulting continuity depends on the spline degree and knot multiplicity. LERP and SLERP are simple $O(1)$ methods with C^0 continuity. SQUAD is also $O(1)$ but achieves C^1 continuity. Bézier and B-Spline methods offer higher-order continuity (up to C^{n-1} under uniform knot spacing) at the cost of increased computational complexity, depending on the degree and number of control points.

Table 4.7 summarizes complexity and continuity. LERP/SLERP are C^0 (continuous position) and the cheapest; SQUAD reaches C^1 continuity; Bezier and B-splines enable higher-order smoothness at higher computational cost. Since interpolation occurs during augmentation, the extra overhead is negligible at inference time, yet the added smoothness/variability improves training quality.

Direct solving vs labeling: A practical question is whether to (i) label markers first (assign identities, then solve) or (ii) perform *direct* landmark denoising and

solving in one model. Label-then-solve pipelines [150, 151] can be effective under clean conditions, but they are prone to error propagation: small identity mistakes (swaps, misses) amplify during skeletal fitting, especially with sparse views or occlusions. Our direct approach integrates denoising, multi-view heatmap fusion, and landmark estimation in a single CNN (Section 4.2.3), regularized by the latent prior and balanced training. Empirically, this reduces brittleness in the presence of ghosting/occlusion and yields higher PCK at tight thresholds (legs/arms on Tail), while also lowering RMSE on balanced sets (cf. Tables 4.1 to 4.3).

In short, end-to-end solving diminishes intermediate failure modes and better exploits prior knowledge—translating to improved robustness without adding latency or post-hoc cleanup.

4.4 Conclusions

Motion-capture datasets exhibit pronounced class imbalance. In this chapter, we introduced a representation-learning strategy for *imbalanced regression* tailored to such data. Our study showed that high-order interpolation schemes generate more useful tail samples than conventional SLERP/LERP, thereby improving coverage of rare poses. In parallel, we leveraged Mahalanobis distance as a relevance signal to identify and reweight infrequent configurations, which proved more discriminative than reconstruction-error heuristics. The resulting training procedure is model-agnostic and can be integrated into diverse architectures, broadening its practical utility.

We expect these findings to encourage the development of more robust pose-solvers and to facilitate wider adoption of data-driven MoCap processing. Looking ahead, incorporating generative models beyond VAEs—such as GANs—offers a promising avenue for further enhancing realism and diversity of synthesized poses, helping to offset long-tail scarcity. Moreover, extending from pose-space priors to motion-space priors (i.e., operating directly on temporally coherent sequences) could address dynamics and continuity explicitly, potentially yielding additional gains in downstream tasks. Collectively, these directions can refine the state of the art and expand the applicability of learning-based MoCap in animation, VR, and human–computer interaction.

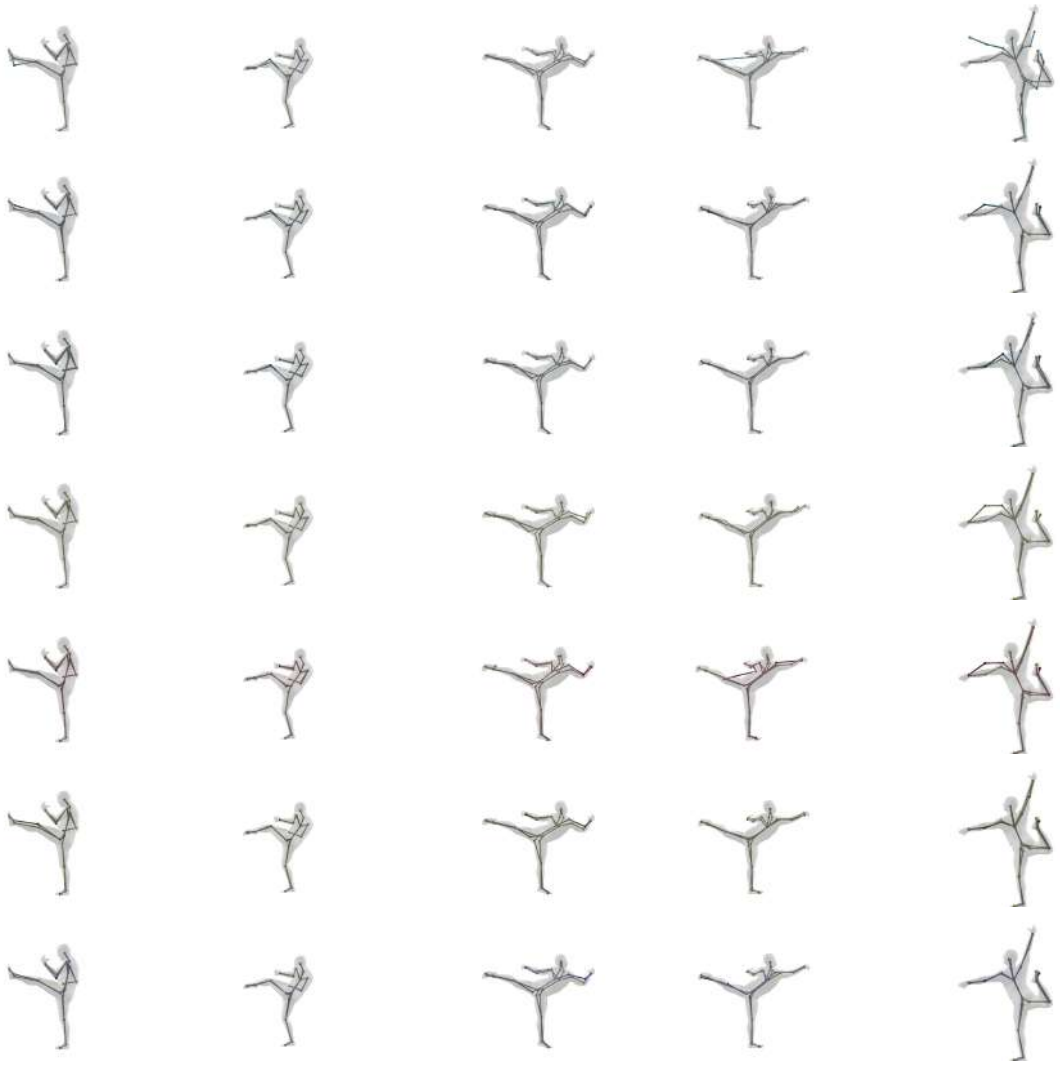


Figure 4.10 Qualitative results of MoCap joint reconstruction. Each joint is visualized in a distinct color, with ground truth shown for reference. The first row presents results from the benchmark method on tail samples, followed by four rows combining the benchmark with our proposed sampling techniques: SLERP, SQUAD, B-spline, and Bezier. The subsequent row illustrates the effect of the relevance reweighting scheme based on a sigmoid function and the Mahalanobis distance error, while the final row demonstrates the synergistic combination of both sampling and reweighting (termed the orthogonality results). All sampling methods enhance the benchmark performance, with higher-order interpolations achieving superior results compared to SLERP. Interestingly, the reweighting scheme influences predictions differently than sampling alone, motivating further investigation into their combined effects.

Chapter 5

Uncertainty-Based Neural Solver

This thesis chapter originally appeared in the literature as [7]

In the previous chapter, we introduced a representation-learning strategy to mitigate bias in human-pose datasets by synthesizing novel samples. We demonstrated the effectiveness of our approach for marker labeling, although noise may persist, affecting the quality of the output motion. Mislabeled markers, occlusions, ghosting artifacts, sensor measurement noise in real-world scenarios, and inference noise from the previously introduced AI model all degrade the performance of existing MoCap solvers. These solvers typically assume a simple isotropic Gaussian noise, identical across all markers/joints, which fails to capture the complex, structured noise present in practice.

In this chapter, we introduce a novel method that addresses the remaining challenge of robustness in the face of the obstacles mentioned above. Optical MoCap systems—particularly those employing sparse, low-cost sensors—produce inherently noisy observations, and learning-based components introduce additional inference variance. The interaction of these noise sources disrupts traditional model-fitting pipelines. Conventional approaches rely on objectives that assume a uniform noise level across the spatial domain (e.g., mean-squared error). Yet, real-world measurements exhibit heteroscedastic, input-dependent noise (e.g., different sensor types may show various types of noise). As a result, these assumptions often lead to degraded solving performance due to the disproportionate influence of noise-induced outliers.

To address this, we propose a noise-aware optimization framework that explicitly models compounded uncertainty to enable robust, marker-based MoCap. Inspired by heteroscedastic uncertainty estimation, the method adaptively reduces the influence of unreliable measurements during body fitting. The proposed approach jointly esti-

mates body-model parameters and a per-landmark uncertainty term, allowing the system to dynamically down-weight inputs affected by sensor or inference noise. This prevents spurious gradients—arising from simplified homoscedastic assumptions—from misleading the optimization, resulting in more accurate displacement estimation.

We evaluate our approach against existing methods, both on controlled datasets and in a real-world setting, demonstrating performance gains in both settings.

5.1 Introduction

This chapter extends the contribution of Chapter 4, which demonstrated how **representation learning** reduces dataset bias and enables efficient, data-driven marker labeling. Once 3D marker positions have been estimated and associated with their corresponding body parts, the next step is to fit a parametric human body model, such as SMPL [27], to these points. This *model-based solving* stage recovers the underlying skeletal structure and detailed body motion, enabling downstream applications such as animation, biomechanics, and virtual reality.

Traditional frameworks, such as MoSh [1], formulate this problem by minimizing a multiterm objective with the data term being the sum of squared distances between observed landmarks and their corresponding model surface points. These objectives implicitly assume additive, *homoscedastic* Gaussian noise ($\epsilon \sim \mathcal{N}(0, \sigma^2)$), thereby treating all observations as equally reliable.

While this assumption holds in controlled studio environments with high-precision, multi-view camera setups, it fails in consumer-grade MoCap systems that rely on sparse, low-cost sensors. Such configurations are subject to high environmental noise, limited signal quality, occlusion, etc. Additionally, the learning-based components introduced in the previous chapter for automating labeling, the markers contribute their own inference uncertainty. Together, these factors produce a noise profile that is highly non-uniform, heavy-tailed, and temporally unstable—ultimately causing traditional solvers to overfit noisy observations, resulting in jittery or physically implausible poses.

Robust estimators, such as Huber[17] or Geman–McClure losses [18], have been proposed to mitigate the influence of outliers by down-weighting large residuals. However, these methods employ fixed-form influence functions that do not adapt to the input noise and require prior confidence knowledge about the measurements.

To address these limitations, we introduce an **adaptive optimization scheme**

based on the Barron loss [19], and further reinterpret robustness as a *learnable* component of the fitting framework. Inspired by the uncertainty-aware likelihood formulation of Kendall [187], as well as by multi-task optimization approaches [187, 188], we jointly estimate body-model parameters and per-landmark uncertainty. This allows the optimizer to automatically down-weight unreliable observations, even without requiring confidence estimation from the AI model, leading to a self-calibrating, noise-aware fitting system that remains stable across heterogeneous sensing environments.

We evaluate the proposed framework on both synthetic and real-world datasets, demonstrating significant improvements in robustness and accuracy compared to the state-of-the-art solvers. Finally, we deploy the method in a real-time marker-based MoCap system using three consumer-grade depth sensors, showcasing its practical effectiveness in challenging, uncontrolled scenarios.

5.2 Methodology

5.2.1 Types of Uncertainty in Low-Cost Optical MoCap

Noise in low-cost optical MoCap systems is inherently heteroscedastic and strongly input-dependent, often deviating significantly from the assumptions underlying classical optimization pipelines. Following the taxonomy introduced by Kendall and Gal [187], the overall error profile can be understood as the interaction between two distinct forms of uncertainty: *aleatoric* and *epistemic*.

Aleatoric uncertainty refers to noise intrinsic to the sensing process and therefore irreducible by simply collecting additional data. In inexpensive RGB-D or multi-view RGB setups, this uncertainty typically manifests as signal-dependent measurement variance that increases with the distance of the subject from the camera or with surface inclination relative to the imaging sensor, or even environmental factors. Additional contributions arise from motion blur, rolling-shutter distortions, illumination fluctuations, and hardware limitations that degrade the signal-to-noise ratio. Occlusions further exacerbate aleatoric noise by causing intermittent missing or corrupted observations, particularly in dynamic sports or clinical environments where body parts frequently self-occlude.

Epistemic uncertainty, by contrast, stems from the model's limited knowledge about the underlying data distribution and therefore reflects uncertainty in the inference process rather than in the physical measurements. In landmark prediction networks, epistemic uncertainty is typically higher at the extremities—such as wrists

and ankles—where visual cues are less stable and multi-view constraints are weaker. It also increases sharply under occlusion, fast motion, and adverse viewpoints, which introduce ambiguities that challenge the model's learned priors. Importantly, this form of uncertainty is reducible: improved training datasets, more expressive model architectures, and diverse capture conditions can all mitigate epistemic uncertainty by strengthening the model's representation of the underlying motion manifold. Our architectural choices in Section 4.2.3 could alleviate some of this uncertainty, but residual epistemic noise remains inevitable in real-world scenarios.

In practical low-cost MoCap systems, these two sources of uncertainty interact in complex ways, giving rise to error patterns that are non-stationary, time-varying, and often heavy-tailed. Reliability varies spatially across the body, with certain joints consistently exhibiting higher predictive variance, while outlier patterns emerge unpredictably as a consequence of both sensor limitations and model ambiguity. The compounded nature of this noise profile poses significant challenges for downstream optimization, as it violates the homoscedastic Gaussian assumptions typically assumed by least-squares formulations, and remains difficult to accommodate even with fixed-form robust estimators. These limitations motivate the development of adaptive, uncertainty-aware optimization strategies capable of modulating their behaviour according to both the spatial structure and temporal dynamics of the underlying noise.

5.2.2 Noise-Aware Optimization

Given a set of estimated 3D landmarks $\boldsymbol{\ell}_{est} \in \mathbb{R}^{J\times 3}$, with J denoting the number of landmarks, the objective is to fit a parametric human body model (e.g., SMPL) by estimating the pose parameters $\boldsymbol{\theta}$, shape coefficients $\boldsymbol{\beta}$, and global transformation $\mathbf{T}$. Traditional frameworks such as MoSh [1] and MoSh++ [2] jointly optimize body parameters and marker configurations, while [189] aims to learn uncertainties during training. In contrast, our method adopts the fixed 53-marker layout introduced in Chapter 4, allowing us to focus exclusively on increasing robustness to unreliable 3D landmark estimates.

As demonstrated in the previous chapter, landmarks inferred from low-cost optical MoCap systems exhibit heteroscedastic, input-dependent uncertainty arising from both sensor-level disturbances and predictive variance in the learned regressor. This induces a noise structure that varies across body parts and frames, violating the homoscedasticity assumptions underlying classical least-squares formulations, which treats all observations as equally reliable. Accurate model fitting, therefore requires an ability to modulate the influence of each landmark, attenuating the effect of uncertain measurements while preserving strong constraints from stable anatomical anchors.

To address this challenge, we embed uncertainty modeling directly within the fitting objective. The optimization jointly estimates the body-model parameters and a set of learned per-landmark reliability weights, yielding a data-driven, noise-adaptive formulation capable of responding to heterogeneous and time-varying noise patterns.

Probabilistic Formulation

Following the uncertainty-aware multi-task learning formulation of Kendall and Gal [187], we model each estimated landmark as an independent isotropic Gaussian random variable with its own variance σ_i^2. Under this assumption, the likelihood of the observed landmarks is

$$p(\boldsymbol{\ell}_{\text{est}} \mid \boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \boldsymbol{\sigma}) = \prod_{i=1}^{J} \mathcal{N}\big(\boldsymbol{\ell}_{\text{est},i} \,\big|\, \boldsymbol{\ell}_i^*(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}), \sigma_i^2 \mathbf{I}\big)\,.$$

Taking the negative log-likelihood yields

$$\mathcal{E}_{\text{data}}(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \boldsymbol{\sigma}) = \sum_{i=1}^{J} \left(\frac{1}{2\sigma_i^2} \left\| \boldsymbol{\ell}_{\text{est},i} - \boldsymbol{\ell}_i^*(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}) \right\|_2^2 + \log(\sigma_i^2) \right) + \text{const}, \tag{5.1}$$

where the constant term does not affect optimization and is omitted henceforth.

For numerical stability and to enforce positivity of the variances, we reparameterize each σ_i^2 in terms of a free scalar $s_i \in \mathbb{R}$ as

$$\sigma_i^2 = \exp(s_i), \qquad \sigma_i = \exp\big(\tfrac{1}{2} s_i\big)\,. \tag{5.2}$$

Substituting into (5.1) gives

$$\mathcal{E}_{\text{data}}(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s}) = \frac{1}{2} \sum_{i=1}^{J} \left(\exp(-s_i) \left\| \boldsymbol{\ell}_{\text{est},i} - \boldsymbol{\ell}_i^*(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}) \right\|_2^2 + s_i \right), \tag{5.3}$$

where $\mathbf{s} = (s_1, \ldots, s_J)^{\mathsf{T}}$ is the vector of log-variances. In this form, $\exp(-s_i) = 1/\sigma_i^2$ acts as a learned precision (inverse variance) that down-weights unreliable landmarks, while the s_i term regularizes the trivial solution of sending variances to infinity.

We regularize the body model using standard SMPL priors [1, 2, 4]:

$$\mathcal{E}_{\text{prior}}(\boldsymbol{\theta}, \boldsymbol{\beta}) = \lambda_\beta \|\boldsymbol{\beta}\|_2^2 + \lambda_{\text{pose}} \|\mathbf{z}\|_2^2, \tag{5.4}$$

where λ_β and λ_{pose} are used for balancing the two terms, and $z = \mathcal{E}(\boldsymbol{\theta})$.

Optionally, we introduce a weak prior on the log-variances to avoid degenerate uncertainty estimates:

$$\mathcal{E}_{\text{unc}}(\mathbf{s}) = \lambda_\sigma \, \|\mathbf{s}\|_2^2, \tag{5.5}$$

with $\lambda_\sigma \geq 0$ controlling the strength of uncertainty regularization.

Assuming independence between parameters and uncertainties, the posterior over $(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s})$ factorizes as

$$p(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s} \mid \boldsymbol{\ell}_{\text{est}}) \propto p(\boldsymbol{\ell}_{\text{est}} \mid \boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s}) \, p(\boldsymbol{\theta}) \, p(\boldsymbol{\beta}) \, p(\mathbf{s}), \tag{5.6}$$

and the maximum a posteriori (MAP) estimate is obtained by minimizing the negative log-posterior, corresponding to

$$\mathcal{E}_{total}(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s}) = \mathcal{E}_{\text{data}}(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s}) + \mathcal{E}_{\text{prior}}(\boldsymbol{\theta}, \boldsymbol{\beta}) + \mathcal{E}_{\text{unc}}(\mathbf{s}). \tag{5.7}$$

Substituting (5.3), (5.4) and (5.5) into (5.7) yields the final objective:

$$\begin{aligned} \mathcal{E}_{total}(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s}) = \frac{1}{2} \sum_{i=1}^{J} \left(\exp(-s_i) \, \|\boldsymbol{\ell}_{\text{est},i} - \boldsymbol{\ell}_i^*(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T})\|_2^2 + s_i \right) \\ + \lambda_\beta \|\boldsymbol{\beta}\|_2^2 + \lambda_{\text{pose}} \|\mathbf{z}\|_2^2 + \lambda_\sigma \|\mathbf{s}\|_2^2. \end{aligned} \tag{5.8}$$

In practice, we minimize $\mathcal{E}_{total}$ jointly with respect to $(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s})$, yielding an uncertainty-aware estimator in which landmark reliabilities and body pose are co-estimated in a single optimization.

5.2.3 Optimization

As such, we express the optimization problem as:

$$\underset{\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{T}, \mathbf{s}}{\operatorname{argmin}} \; \mathcal{E}_{\text{data}} + \mathcal{E}_{\text{prior}} + \mathcal{E}_{\text{unc}} \tag{5.9}$$

.

To ensure stable convergence, we adopt a two-stage optimization strategy inspired by MoSh++. In the first stage, we jointly optimize $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, and $\mathbf{T}$ using fixed uncertainty values, yielding a coarse but stable initialization. In the second stage, $\boldsymbol{\beta}$ is held fixed while $\mathbf{T}$, $\boldsymbol{\theta}$ and the log-variance vector $\mathbf{s}$ are refined jointly, as depicted in Figure 5.1. This allows the solver to reassess the reliability of each landmark and to down-weight uncertain extremities or occluded joints.

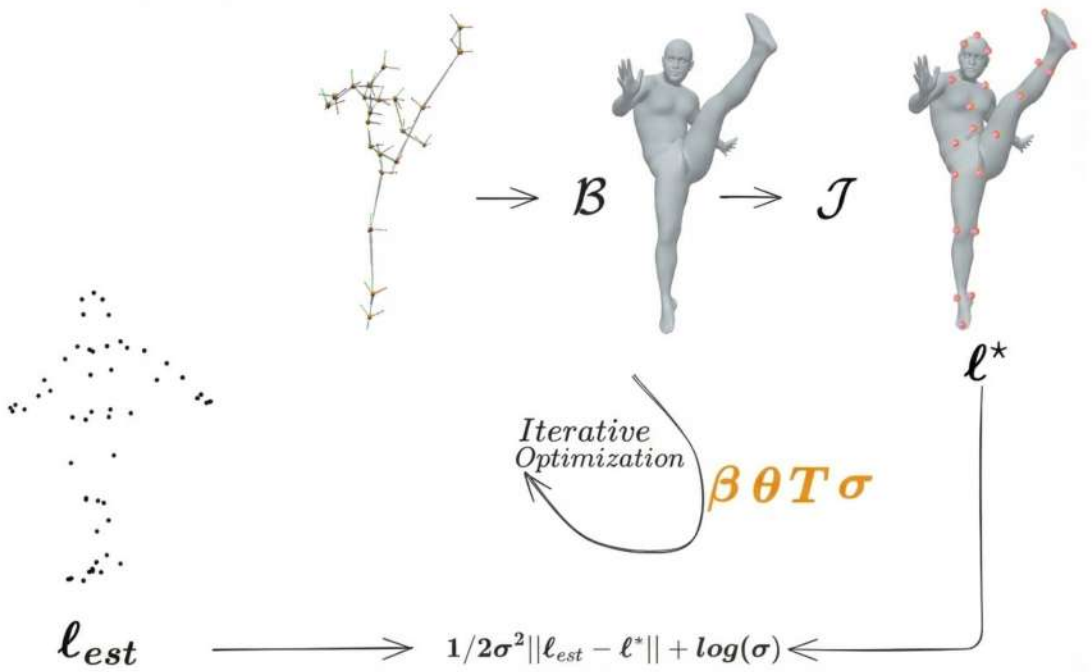


Figure 5.1 Overview of our uncertainty-aware body-model fitting pipeline. The input consists of the estimated 3D landmarks $\boldsymbol{\ell}_{\text{est}}$ (black dots, bottom-left). From left to right, the optimization predicts $\boldsymbol{\beta}$, $\boldsymbol{\theta}$, $\mathbf{T}$ (shape, pose, and global transformation). These are passed through the body model $\mathcal{B}$ and regressor $\mathcal{J}$ to obtain the fitted mesh and its corresponding landmarks $\boldsymbol{\ell}^\star$ (grey body). Per-landmark uncertainties $\boldsymbol{\sigma}$ are visualized as colored markers on the final mesh (top-right). The bottom path shows the heteroscedastic data term $\frac{1}{2\sigma^2}\|\boldsymbol{\ell}_{\text{est}} - \boldsymbol{\ell}^\star\|^2 + \log(\sigma^2)$, whose gradient (curved arrow) drives the iterative update of both the model parameters and the uncertainty estimates.

We argue that the learned uncertainties reshape the loss landscape dynamically: stable landmarks dominate the early iterations, whereas unreliable measurements are progressively attenuated. Large $\mathbf{s}$ flattens the curvature of the data term around uncertain landmarks, reducing their gradient magnitude and preventing overfitting. Conversely, stable landmarks with low uncertainty dominate the descent direction. This produces a dynamically reweighted loss landscape that evolves during optimization, improving convergence under heavy-tailed noise. We prove that this approach yields smoother and more physically plausible pose estimates and prevents overfitting to noisy observations, a common failure mode in low-cost capture setups.

5.3 Evaluation

To rigorously assess the solver's resilience to the heavy-tailed noise distributions characteristic of low-cost sensors, we conducted a controlled quantitative evaluation on the THuman2.0 dataset [184], while also demonstrating the effectiveness of our approach in a real-time capture system using consumer-grade depth sensors. The proposed system has also been demonstrated in a real-time demo scenario in ICCV 2023 [24].

5.3.1 Synthetic Evaluation

We utilize the THuman2.0 dataset [184], which provides accurate SMPL fits of human subjects in diverse poses. From these scans, we synthesize 3D marker observations by placing virtual markers on the surface according to the layout defined in Chapter 4. This allows us to generate ground-truth landmark positions for quantitative evaluation. To simulate the compounded uncertainty present in real-world low-cost MoCap systems, we inject two types of synthetic noise into the ground-truth markers, and finally combine both of them:

- **Data Noise (n_d):** We model noise arising directly from the input markers through three corruption mechanisms:

  1. *Occlusion.* Let $\mathbf{p} = (\mathbf{p}_1, \ldots, \mathbf{p}_n)$ denote the marker positions. We randomly mask k markers, where $k \sim \mathcal{U}(m,\, n')$ with $m \leq n' \leq n$ and m and n' define the minimum and maximum number of occluded markers.
  2. *Ghost markers.* Following [151], we fit a Gaussian to the remaining markers by computing per-axis medians $\boldsymbol{\mu}$ and the sample covariance Σ, then draw synthetic positions $g \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ and append them to $\mathbf{p}$.
  3. *Measurement noise.* We select N markers at random and perturb each with an offset $o \sim \mathcal{U}(-M_{noise},\, M_{noise})$, yielding corrupted positions $\mathbf{p}' = \mathbf{p} + \mathbf{o}$.

- **Inference Noise (n_m):** To simulate the uncertainty introduced by the automatic labeling using AI, we also incorporate model-driven inference noise. Specifically, we use the model described in Section 4.2.3, trained following the procedure outlined in the previous chapter, and treat its prediction variability as an additional noise source.

To prove the effectiveness of our uncertainty-aware optimization framework, we compare it against two distinct baselines:

- **Standard Optimization:** A MoSh++ style baseline [2] that minimizes an unweighted L_2 loss. This represents the industry standard for marker solving, which implicitly assumes homoscedastic noise.
- **Robust Optimization:** A solver utilizing the adaptive Barron loss [19]. This represents a state-of-the-art robust estimator that attempts to handle outliers via a learnable shape parameter α, but without explicitly modeling input-dependent uncertainty. Essentially, α controls the robustness of the estimator. Different values of α interpolate between quadratic (Gaussian-like) and heavy-tailed robust losses, allowing the loss to adapt to the different kinds of noise.

For a fair comparison with [19], we performed a grid search for its shape parameter α; Figure 5.2 shows RMSE_3 values over $\alpha \in [-7, 4]$, identifying -4.5 as the best-performing initialization. However, this is also a disadvantage of the method, as it requires careful hyperparameter tuning for different noise levels, whereas our approach learns uncertainty directly from the data.

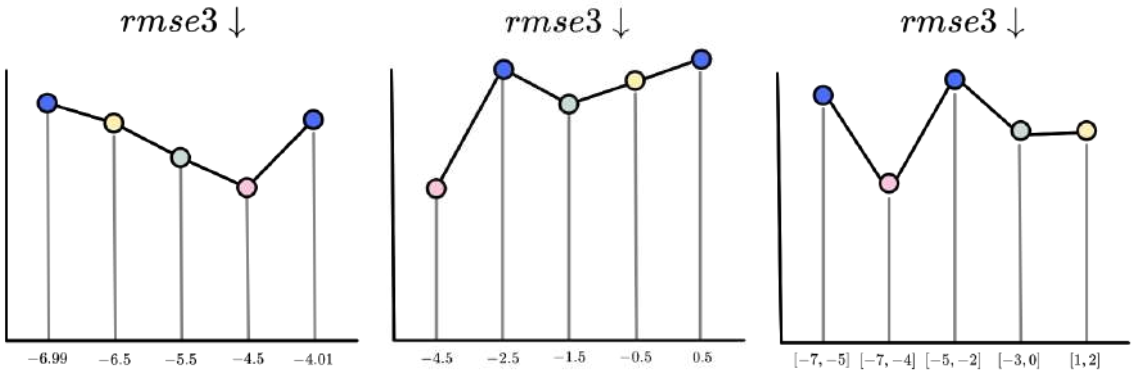


Figure 5.2 Ablation study on the Barron loss shape parameter α, which controls the robustness level of the estimator. In all cases, α is treated as a fixed hyperparameter and selected via grid search. (a) Using $\alpha_{\text{range}} \in [-7, -4]$, we perform a grid search over candidate values and select the best α (denoted α_{init}) based on RMSE_3. (b) Using a wider range $\alpha_{\text{range}} \in [-7, 2]$, we repeat the grid search to evaluate the sensitivity of the method to the chosen interval. (c) Initializing α at the mean of α_{range}, we progressively refine the search interval to identify the best-performing value.

Results and Analysis

The results, summarized in Table 5.1, demonstrate a clear hierarchy of robustness. The standard fitting approach struggles significantly under high noise, as the L_2

penalty forces the model to accommodate outliers, degrading the overall pose alignment. While the Barron loss offers some improvement by mitigating the influence of extreme outliers, we found it challenging to tune; its shape parameter α adds optimization complexity that can lead to instability in the high-dimensional pose space.

In contrast, our **uncertainty-aware formulation** significantly outperforms both baselines in most of the metrics. By jointly learning the variance parameters, our method effectively creates a dynamic "trust" mechanism. It does not merely dampen significant errors universally (like robust losses) but identifies *specific* unreliable landmarks to down-weight, allowing the solver to prioritize high-confidence data points for precise alignment. This confirms that modeling heteroscedasticity is crucial for marker-based solving in uncontrolled environments.

	RMSE ↓	MAE ↓	PCK1 ↑	PCK3 ↑	PCK7 ↑
[1, 2]	30.1 mm	3.49°	11.79%	66.85%	**98.34**%
[19]	30.8 mm	3.10°	12.71%	67.06%	97.71%
Ours ($\boldsymbol{\ell}^m$)	**28.9** mm	**2.98°**	**14.71**%	**69.86**%	98.18%

Table 5.1: Noisy landmark fits comparison on TH2. Uncertainty-aware optimization outperforms baselines in all metrics yielding lower reconstruction error and higher accuracy in terms of PCK.

Beyond numerical metrics, visual inspection offers critical insight into the practical advantages of this approach. Figure 5.6 presents a qualitative comparison between a naive solver and our proposed method using data captured from a sparse, three-sensor setup.

The standard solver (Naive Fit) exhibits characteristic artifacts of homoscedastic assumptions: when a limb is occluded or rushes (inducing motion blur), the corresponding inferred markers become noisy. The naive solver, treating these noisy predictions as ground truth, contorts the body model to reach them, resulting in unnatural limb bending and jitter.

Our method (Uncertainty-Aware Fit) yields a substantially more stable reconstruction. As illustrated in the figure, the model successfully maintains a plausible human pose even when individual markers deviate significantly. This indicates that the optimization has correctly assigned high uncertainty values (σ) to the noisy observations, effectively "ignoring" the sensor artifacts and relying on the learned prior to hallucinate the correct geometry. This ability to filter measurement noise at the optimization level is key to enabling reliable real-time MoCap with consumer-grade hardware.

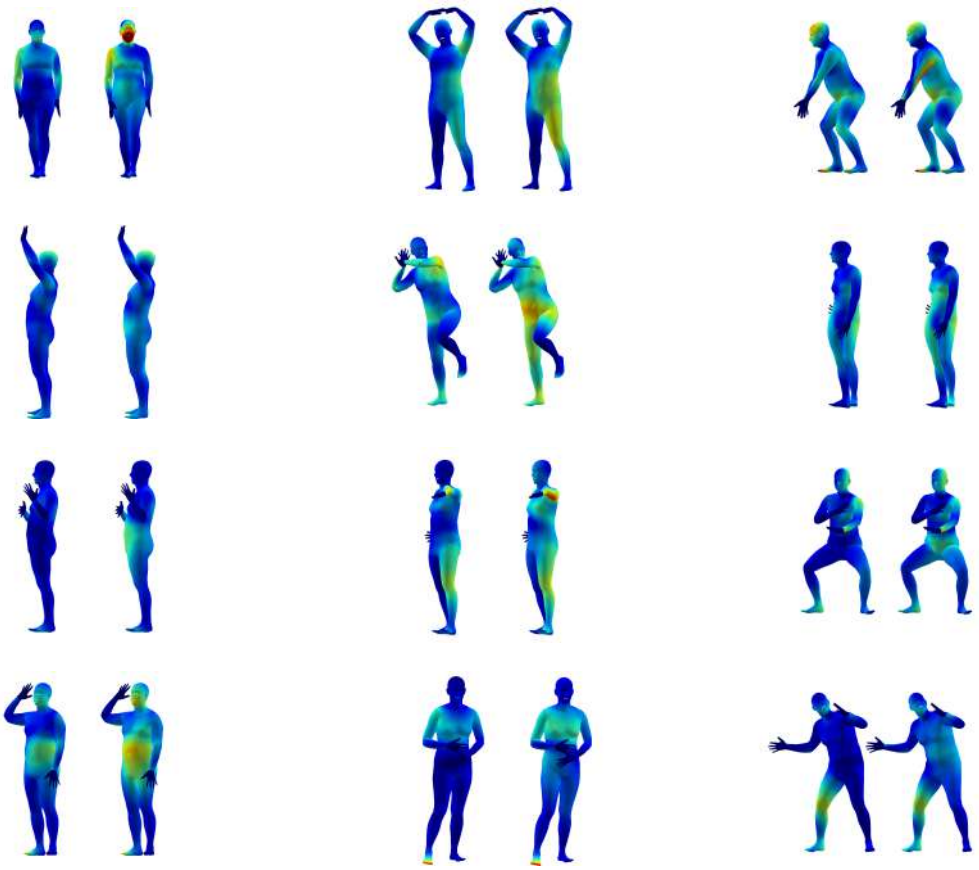


Figure 5.3 Qualitative comparison between our noise-aware fitting method and MoSh [1]. The figure shows results from our approach (*left*) and from [1] (*right*). Each mesh is color-coded using a Jet colormap based on the Euclidean distance error from the ground-truth mesh, where warmer colors indicate higher reconstruction error. Clearly the noise-aware fitting produces more accurate results, especially in challenging areas such as the hands and feet.

	n_d	n_m	RMSE ↓	MAE ↓	PCK1 ↑	PCK3 ↑	PCK7 ↑
[1, 2]	✓	✗	30.10 mm	3.49°	11.79%	66.85%	98.34%
[19]			30.80 mm	3.10°	12.71%	67.06%	97.71%
Ours ($\boldsymbol{\ell}^m$)			28.90 mm	2.98°	14.71%	69.86%	98.18%
Ours ($\boldsymbol{\ell}^m \vert \boldsymbol{\ell}^j$)			**23.40** mm	**2.29°**	**19.66%**	**81.06%**	**99.11%**
[1, 2]	✗	✓	20.60 mm	1.93°	28.71%	89.03%	**99.05%**
[19]			21.71 mm	1.91°	36.38%	87.75%	98.22%
Ours ($\boldsymbol{\ell}^m$)			18.70 mm	1.85°	41.99%	90.95%	98.81%
Ours ($\boldsymbol{\ell}^m \vert \boldsymbol{\ell}^j$)			**18.50** mm	**1.49°**	**42.18%**	**91.44%**	98.56%
[1, 2]	✓	✓	23.80 mm	2.03°	24.26%	85.63%	**98.22%**
[19]			24.87 mm	1.94°	31.99%	84.05%	97.00%
Ours ($\boldsymbol{\ell}^m$)			22.40 mm	1.79°	36.01%	87.14%	97.53%
Ours ($\boldsymbol{\ell}^m \vert \boldsymbol{\ell}^j$)			**21.90** mm	**1.52°**	**36.67%**	**88.09%**	97.69%

Table 5.2: Noisy landmark fitting on THuman 2.0. Comparison of our noise-aware fitting approach vs. a variant of the fitting method from [1, 2]. RMSE is in millimeters (mm) and PCK in (%). Subscripts j and m denote "joints" and "markers", while n_d and n_m indicate data and marker noise, respectively.

Apart from that, we also examine the impact of the different types of noise described earlier. As evident from Table 5.2, our method consistently outperforms the baselines across all noise conditions, demonstrating its versatility and robustness. In contrast, [19] exhibits strong dependence on its hyperparameter configuration. Our approach, by learning $\boldsymbol{\sigma}$ directly removes the need for manual tuning.

We further compare optimization with estimated landmarks $\boldsymbol{\ell}_{est}$ of both markers ($\boldsymbol{\ell}^m$) and joints ($\boldsymbol{\ell}^j$), and how our uncertainty-aware formulation performs under each setting. As shown in Table 5.2, using either markers or joints alone yields improvements over the baselines, with markers providing stronger constraints due to their higher spatial density. Notably, our method benefits from combining both inputs, observing additional performance gains when both are used. Because each landmark's influence is scaled by its learned uncertainty, the system automatically balances marker and joint contributions. Figure 5.3 visualizes vertex-wise reconstruction errors relative to ground truth.

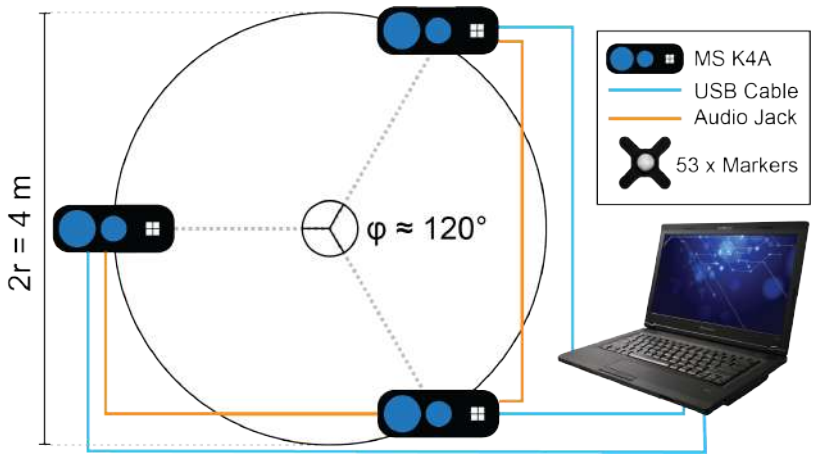


Figure 5.4 The proposed MoCap system comprises hardware (HW) and software (SW) components. From a HW perspective, a minimal set of tripod-mounted commodity sensors are required (3 Microsoft K4A shown), connected with a workstation that handles the processing (cyan links). Typical outwards-in placement requires them to be equidistantly placed from an angular perspective around a pre-determined radius ($r = 2m$ in this case). Additionally, HW synchronization cables inter-connect the sensors (orange links). Finally, 53 retro-reflective markers are also required to be placed onto the subject to be captured.

5.3.2 Real-World Deployment

Additionally, we prove the effectiveness of our approach in real-world settings. Specifically, we deploy it in a real-time, marker-based MoCap system using a sparse set of commodity depth sensors. We construct a multi-sensor capture rig consisting of three Microsoft Azure Kinect (K4A) devices, as shown in Figure 5.4.

Hardware Overview

The system requires only a minimal set of tripod-mounted depth sensors -three K4A units in our setup- all connected to a workstation that performs the processing (cyan links). For outward-in capture, the sensors are placed equidistantly around a circular perimeter with radius $radius = 2\,\text{m}$. Hardware synchronization is achieved using dedicated sync cables (orange links). A total of 53 retro-reflective markers are attached to the subject. Retro-reflective operation requires the hardware to project infrared (IR) light and capture the reflected signal—both features supported by the K4A. To keep the system lightweight and easy to deploy, we intentionally use a small number of sensors (3–6), relying on depth information to compensate for the reduced

viewpoint coverage. Given the discontinuation of the Intel RealSense RS2 line and the pending availability of active-IR OAK-D units, the Azure Kinect offers the most practical choice for our prototype.

Marker Acquisition

Each sensor $s \in \{1, \ldots, N_{cam}\}$ streams synchronized infrared $\mathbf{I}_s(\mathbf{px}) \in \mathbb{R}$ and depth $\mathbf{D}_s(\mathbf{px}) \in \mathbb{R}$ frames at each pixel $\mathbf{px} = (u, v) \in \Omega$ with Ω denoting an image of resolution $H \times W$. Both signals are 16-bit and pixel-aligned.

Retro-reflective markers appear as saturated bright regions in $\mathbf{I}_s$ due to their high return amplitude (see Figure 5.5, top). They are extracted via simple IR thresholding.

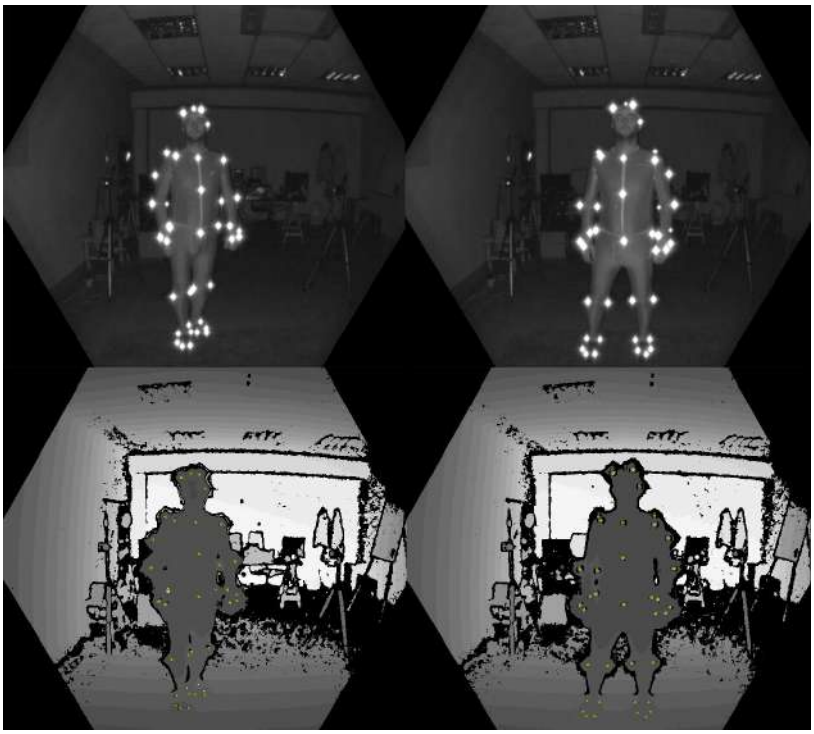


Figure 5.5 Two representative examples of marker detection using the K4A sensor. The top row shows the corresponding IR frames, where the retro-reflective markers appear as bright, easily distinguishable points. In the bottom row, the detected marker locations (shown as green stars) are projected onto the depth images. These detections fall within regions of missing or invalid depth—areas saturated by the retro-reflective returns—creating characteristic "blind" patches in the K4A depth measurements.

Thresholding yields $N_{markers}$ marker blobs represented by their centroids $\boldsymbol{\mu}_n^s \in \Omega$.

Although the IR amplitude saturates at the centroid, preventing reliable depth at that pixel, the spherical markers scatter IR light around their perimeter, producing a depth "ring" that approximates the marker surface. We take the median depth in this ring to obtain a denoised, unbiased estimate, which is then lifted to 3D using the sensor intrinsics $\mathbf{K}_s$ and distortion coefficients $\mathbf{d}_s$, resulting in 3D marker estimates $\mathbf{m}_n^s \in \mathbb{R}^3$. The detection process is illustrated in Figure 5.5 (bottom).

Spatiotemporal Sensor Calibration

Accurate multi-view fusion requires all local 3D marker estimates $\mathbf{m}_n^s$ to be aligned both temporally and spatially.

Temporal Alignment. We use the K4A's hardware synchronization and apply a small timing offset of $160\mu s$[1] to mitigate multi-path interference. This ensures reliable temporal alignment for dynamic motion sequences.

Spatial Alignment (Wand Calibration). To recover each sensor's 6DoF pose

$$\mathbf{T}^s = \begin{bmatrix} \mathbf{R}_s & \mathbf{t}_s \\ \mathbf{0}^\top & 1 \end{bmatrix},$$

we perform a lightweight wand-based calibration:

1. A single retro-reflective marker attached to a wand is moved in front of all sensors.
2. When all sensors detect exactly one marker, we record the corresponding 2D detections $\boldsymbol{\mu}_k^s$ and 3D estimates $\mathbf{m}_k^s$.
3. We estimate initial extrinsics via pairwise Umeyama alignment [190] relative to a reference sensor $s_{\text{ref}} = 1$.
4. We refine these extrinsics using sparse bundle adjustment [191] with the 2D projections as constraints, keeping the reference pose fixed.

The resulting extrinsics $\mathbf{T}^s$ allow us to fuse all marker estimates into a unified 3D marker cloud:

$$\hat{\mathbf{m}} = \bigcup_{s=1}^{N_{cam}} \bigcup_{n=1}^{N_{markers}} \mathbf{T}^s \mathbf{m}_n^s.$$

[1] Following the official K4A documentation.

We then perform gravity alignment and radius-based clustering to correct small inconsistencies. Because only $N_{cam} = 3$ viewpoints are used, the fused cloud contains higher uncertainty, precisely the setting in which our noise-aware solver proves most beneficial.

Final Processing Pipeline

The fused marker cloud is fed into the model described in Section 4.2.3 for automatic labeling. The resulting structured landmark set (i.e., $\boldsymbol{\ell}_{\text{est}}$) is then provided as input to our uncertainty-aware solver, which robustly estimates the body model parameters even under noisy, incomplete, or partially occluded observations.

Qualitative Results

We demonstrate that the proposed real-time capture system produces stable, high-quality results under sparse sensing. Qualitative examples are presented in Figure 5.7. A real-time demonstration video is also available.[2]

Under these settings, the benefits of uncertainty-aware optimization become even more pronounced. Commodity depth sensors frequently produce intermittent dropouts, IR-saturation artefacts, and short-lived geometric inconsistencies due to motion blur or rapid viewpoint changes. Classical solvers must either over-smooth the solution to remain stable or attempt to fit these erroneous observations, resulting in inconsistent pose estimates. In contrast, the proposed formulation automatically down-weights unreliable markers on a frame-by-frame basis through the learned σ_i values, preserving responsiveness without sacrificing stability.

5.4 Conclusions

This chapter addressed the challenge of compounded uncertainty in marker-based motion capture, arising from both sensor noise and model inference variance. We proposed a *noise-aware optimization framework* that generalizes classical fitting methods by treating uncertainty as a learnable, per-landmark parameter. Through this mechanism, the model autonomously estimates the reliability of each observation and adapts

[2]Real-time system demonstration video.

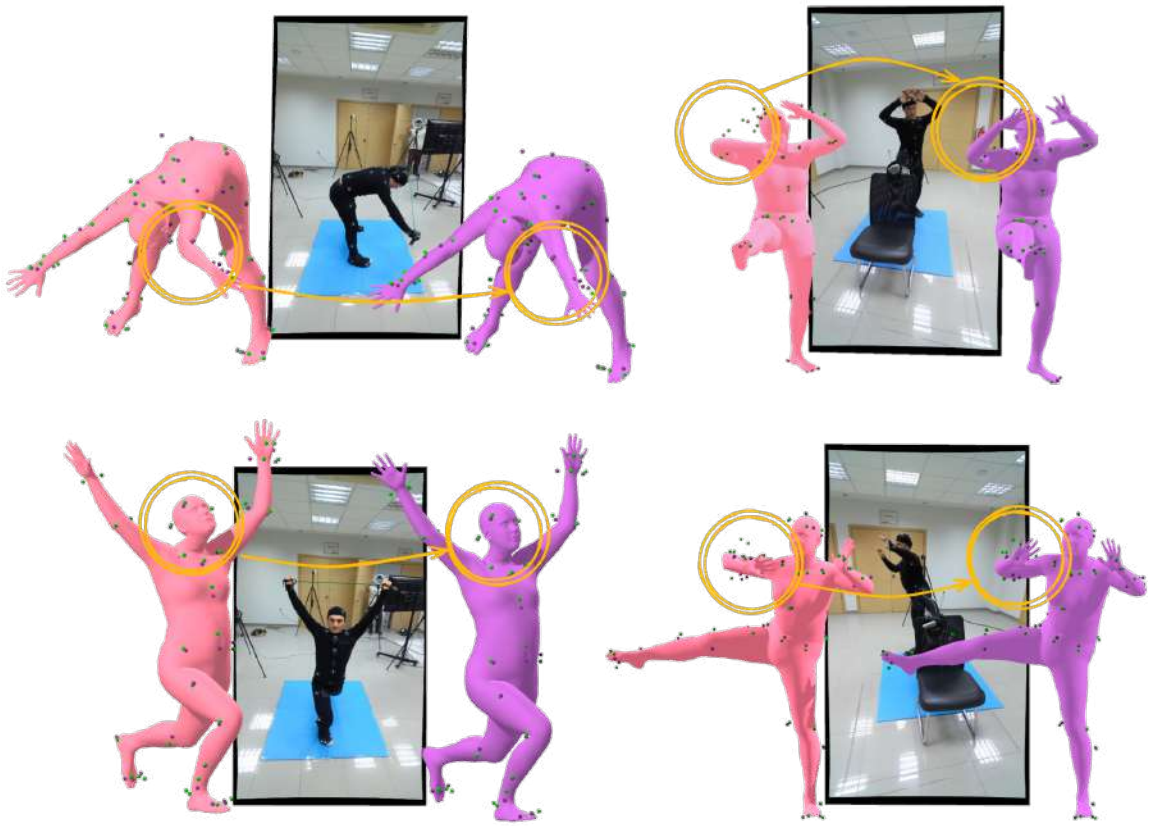


Figure 5.6 Plain vs. uncertainty-based fitting. Input markers from the consumer-grade system and the model-inferred markers are shown in green and violet, respectively. The uncertainty-aware fit yields smoother and more consistent marker alignments compared to the plain baseline.

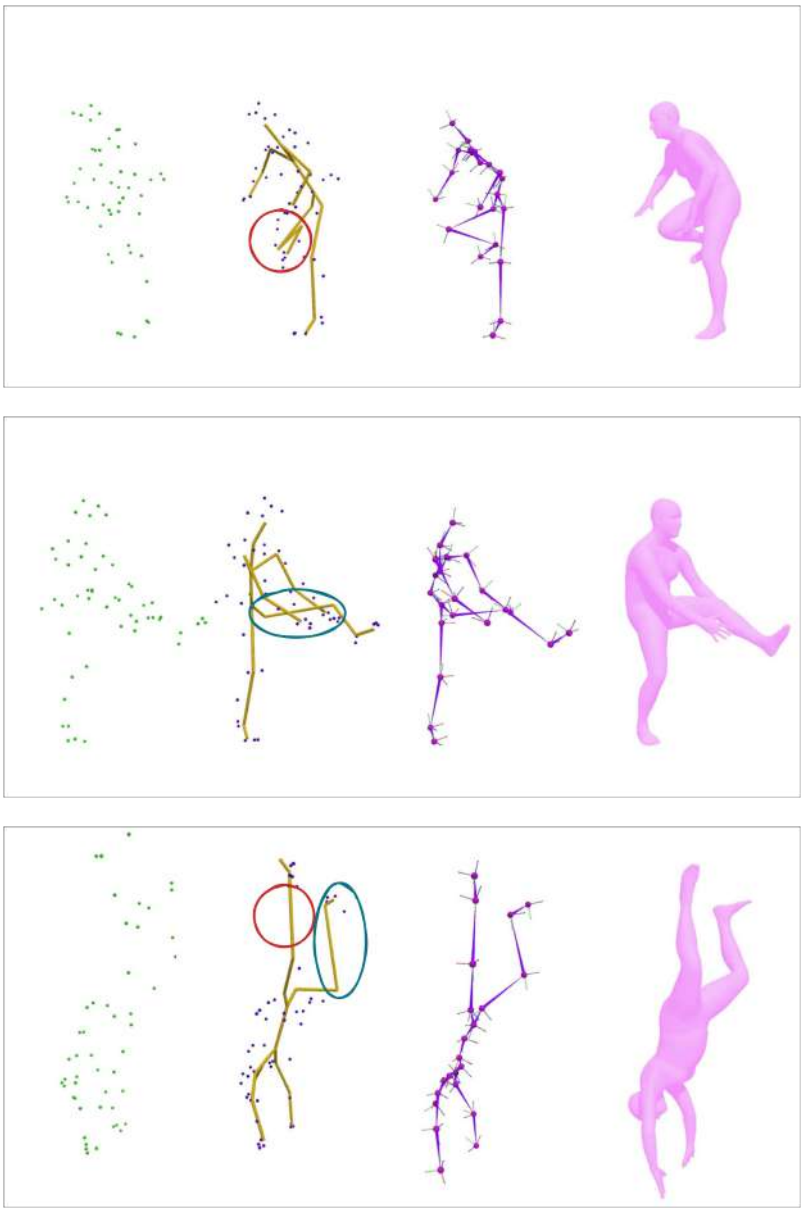


Figure 5.7 Additional qualitative results of our system in the wild. Results were obtained using a sparse setup of low-cost sensors. From left to right: raw input captured by the multi-sensor acquisition system (Section 5.3.2); unfiltered estimated landmarks $\boldsymbol{\ell}_{est}$ produced by our model; and the final fitted pose $\boldsymbol{\theta}_{est}$ and shape $\boldsymbol{\beta}_{est}$ parameters. Since the real-time model implicitly learns the human skeleton, it may occasionally yield unrealistic poses. The proposed noise-aware fitting framework effectively introduces human body constraints, producing more accurate and anatomically plausible results, while also handling missing or misdetected landmarks.

the weighting accordingly, without requiring manual tuning or prior confidence estimates. Experimental results—including real-time deployment "in the wild" with a minimal sensor setup—demonstrate that this approach yields robust, high-fidelity reconstructions even under adverse conditions. Overall, this work advances the goal of accessible, low-cost, real-time optical MoCap and lays the groundwork for the transition to the fully markerless systems explored in the following chapter. A real-time demo of the system has been presented at ICCV 2023 [24]. While effective, the proposed method inherits certain limitations. Extremely sparse or systematically missing landmarks may cause the learned uncertainties to saturate, reducing the influence of valid geometric constraints. The Gaussian assumption, although well-suited for local residuals, may under-represent strongly multimodal noise patterns caused by severe occlusions. Finally, the present formulation is frame-based and does not explicitly propagate uncertainty through time.

Chapter 6

Temporally Consistent Body Solver

This thesis chapter originally appeared in the literature as [21] and [10]

Generating temporally consistent and geometrically accurate human motion from sparse videos—without relying on markers or specialized hardware—remains a long-standing challenge in the motion capture community. Many existing methods employ intricate multi-stage pipelines that blend data-driven regression with iterative optimization and temporal regularization, which in turn increases computational cost and require extensive hyperparameter tuning for each objective term.

In contrast, within this chapter, we introduce **BundleMoCap++**, which provides a conceptually simpler yet highly effective alternative. It reconstructs motion in a single optimization stage, dispensing entirely with explicit temporal smoothness constraints while still yielding continuous, stable results. This approach not only attains state-of-the-art accuracy but also reduces overall complexity. The method relies on interpolating multiple latent keyframes under a local manifold-smoothness assumption, using efficient interpolation schemes to jointly recover a bundle of frames from two or more latent codes. Implemented as a sliding-window optimizer, it requires initialization from only the first frame, which further limits computational overhead.

The strength of BundleMoCap++ lies in achieving high-quality reconstructions using modest resources. We introduce a novel hyperspherical human-pose prior that geometrically constrains the latent space, enabling more expressive interpolation. An additional contribution is a direct optimization algorithm operating on the learned manifold, improving convergence stability and accuracy. Finally, we extend traditional interpolation techniques to higher-order forms adapted for hyperspherical manifolds, allowing larger temporal windows to be solved efficiently while preserving

motion realism.

6.1 Introduction

While the uncertainty-aware optimization framework proposed in the previous chapter (Chapter 5) proved effective for noisy, marker-based motion data, its principles are insufficient for the significantly more complex domain of **markerless motion capture (MoCap)** [192]. Transitioning from physically measured markers to image-inferred keypoints introduces a cascade of new difficulties that demand a fundamentally different modeling paradigm (Section 1.2).

A central challenge is the lack of **temporal coherence**. Purely data-driven monocular methods often process frames independently, leading to discontinuous and implausible motion reconstructions [15, 193–196]. The problem is even more pronounced in sparse multi-view setups, where data scarcity makes direct end-to-end learning impractical. As a result, many state-of-the-art systems rely on optimization frameworks that impose explicit smoothness terms across temporal windows [5, 6, 8, 9].

However, these pipelines inherit fragility from their primary data source—2D keypoints generated by detectors such as [69]. Such detections are not only noisy and jitter-prone but often contain severe artifacts including missing limbs or flipped body parts [197]. To mitigate these issues, previous research has introduced elaborate, multi-stage optimization strategies that iteratively refine and filter erroneous initial estimates [5, 8, 9, 99]. These methods, though effective, come at the expense of runtime and parameter complexity.

We present a new formulation that resolves these limitations within a single, streamlined stage. Instead of optimizing a large batch of frame-specific variables, our method estimates only two **latent codes** whose **manifold interpolation** reconstructs an entire temporal bundle. By operating directly in this low-dimensional latent domain, the method achieves efficiency and smoothness simultaneously, producing realistic motion trajectories without any explicit temporal-regularization loss.

To make this possible, we introduce several technical advances: (1) a hyperspherical pose prior that defines a geometrically coherent latent manifold, (2) an optimization algorithm that operates directly on this manifold, enabling consistent updates along geodesic directions, and (3) higher-order interpolation schemes designed for hyperspherical spaces, which allow broader temporal windows to be solved more effectively. Together, these components form the foundation of **BundleMoCap++**,

a framework that delivers temporally consistent, high-fidelity motion capture with minimal computational overhead.

6.2 Approach

6.2.1 Preliminaries

We employ a parametric human body model $\mathcal{B}$ that maps parameters to a body geometry $(\mathbf{v}, \mathbf{f}) = \mathcal{B}(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{T})$. The triangular surface $(\mathbf{v}, \mathbf{f})$ is defined by vertices $\mathbf{v} \in \mathbb{R}^{V\times 3}$ and faces $\mathbf{f} \in \mathbb{N}^{F\times 3}$, with V and F the counts of vertices and faces. The model synthesizes shape via blendshape coefficients $\boldsymbol{\beta} \in \mathbb{R}^{10}$, articulates joints with pose parameters $\boldsymbol{\theta} \in \mathbb{SO}(3)^P$, and places the mesh globally with $\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{SE}(3)$. Linear regression with $\mathcal{J}$ yields joints $\boldsymbol{\ell}^j = \mathcal{J}\,\mathbf{v}$. Joints $\boldsymbol{\ell}^j$ are then projected into the image domain $\Omega := \mathbb{R}^{W\times H}$ as keypoints $\mathbf{k} = \boldsymbol{\pi}(\boldsymbol{\ell}^j)$, where $\boldsymbol{\pi}$ is parameterized by the camera intrinsics of Ω.

6.2.2 Spherical Autoencoder

Following [142], we constrain centered latent variables to lie on a unit sphere rather than enforcing a KL alignment $\mathrm{KL}[q(\mathbf{z}|\mathbf{x}) \,\|\, p(\mathbf{z})]$. This allows us to impose the latent-manifold geometry directly. Concretely, we train a VAE to embed valid human poses on a hypersphere $\mathcal{M}$ with $d_z = 32$, optimizing

$$\mathcal{L}_{SAE} = \lambda_1 \mathcal{L}_{rec} + \lambda_2 \mathcal{L}_{orth}, \tag{6.1}$$

subject to

$$\text{s.t.} \quad \mathbf{z}^\top \mathbf{z} - \mathbf{1} = 0, \quad \mathbf{z} \in \mathbb{S}^{d_z - 1}, \tag{6.2}$$

$$\mathcal{L}_{rec} = \|\mathbf{v} - \hat{\mathbf{v}}\|_2^2, \tag{6.3}$$

$$\mathcal{L}_{orth} = \frac{\mathrm{Tr}(\mathbf{R}^\top \hat{\mathbf{R}}) - 1}{2}, \tag{6.4}$$

where $\mathbf{z} \in \mathbb{R}^{32}$ is the latent code, $\mathbf{R} \in \mathbb{SO}(3)^P$ are per-part rotations, $\hat{\mathbf{R}}$ their decoded counterparts, λ_1, λ_2 weighted terms, and $\mathbf{v}, \hat{\mathbf{v}}$ the ground-truth/predicted vertices (so $\mathcal{L}_{rec}$ covers both angular and 3D errors).

To enforce Eq. (6.2), we apply spherical normalization (center–subtract and renormalize) in the latent space:

$$\boldsymbol{\theta} \xrightarrow{\text{encoder}} \mathbf{z} \xrightarrow{\text{spherical constraint}} \frac{(\mathbf{z} - \bar{\mathbf{z}}\,1)}{\|\mathbf{z} - \bar{\mathbf{z}}\,1\|} \xrightarrow{\text{generator}} \tilde{\boldsymbol{\theta}},$$

with

$$\bar{\mathbf{z}} = \frac{1}{d_z} \sum_j \mathbf{z}^j.$$

Instead of variational regularizers or explicit hyperspherical priors [138, 139], this purely geometric construction avoids probabilistic tuning while mapping pose embeddings onto the hypersphere $\mathcal{M}$ directly.

6.2.3 Sider Interpolation

Having trained **SPoser** to embed poses on $\mathcal{M}$, we can exploit high-order *spherical* interpolation schemes. We adopt *Spherical Interpolation of orDER* (SIDER-n) [145], which guarantees intermediates remain on the hypersphere.

A hypersphere $S^n \subset \mathbb{R}^{n+1}$ is

$$S^n = \{\mathbf{x} \in \mathbb{R}^{n+1} \mid \|\mathbf{x}\| = 1\}.$$

With the induced Euclidean metric, the geodesic distance between $\mathbf{q}_1, \mathbf{q}_2 \in S^n$ is

$$d(\mathbf{q}_1, \mathbf{q}_2) = \cos^{-1}(\mathbf{q}_1 \cdot \mathbf{q}_2),$$

and the geodesic $\gamma(t)$ is

$$\gamma(t) = \frac{\sin((1-t)\psi)}{\sin\psi}\mathbf{q}_1 + \frac{\sin(t\psi)}{\sin\psi}\mathbf{q}_2, \quad \psi = d(\mathbf{q}_1, \mathbf{q}_2).$$

The tangent space at $\mathbf{q}$ is $T_{\mathbf{q}}S^n = \{\xi \mid \xi \perp \mathbf{q}\}$, with log/exp maps

$$\log_{\mathbf{q}_1}(\mathbf{q}_2) = \frac{\psi}{\sin\psi}\left(\mathbf{q}_2 - (\mathbf{q}_1 \cdot \mathbf{q}_2)\mathbf{q}_1\right), \qquad \exp_{\mathbf{q}}(\xi) = \cos\|\xi\|\,\mathbf{q} + \sin\|\xi\|\,\frac{\xi}{\|\xi\|}.$$

SLERP is then

$$\mathrm{SLERP}(\mathbf{q}_1, \mathbf{q}_2, t) = \exp_{\mathbf{q}_1}\left(t\,\log_{\mathbf{q}_1}(\mathbf{q}_2)\right).$$

For a spherical *quadratic* curve through $\mathbf{q}_2$ given key points $\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3 \in S^n$ and $t \in [0, 1]$, SIDER2 constructs geodesic control points via extrapolation:

$$\mathbf{d}_{2a} = \mathrm{SLERP}(\mathbf{q}_3, \mathbf{q}_2, 2), \qquad \mathbf{d}_{2b} = \mathrm{SLERP}(\mathbf{q}_1, \mathbf{q}_2, 2),$$

then

$$\mathbf{c}_{inner} = \mathrm{SLERP}(\mathbf{q}_1, \mathbf{d}_{2a}, t), \qquad \mathbf{c}_{outer} = \mathrm{SLERP}(\mathbf{d}_{2b}, \mathbf{q}_3, t),$$

and finally

$$\mathrm{SIDER2}(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, t) = \mathrm{SLERP}(\mathbf{c}_{inner}, \mathbf{c}_{outer}, t).$$

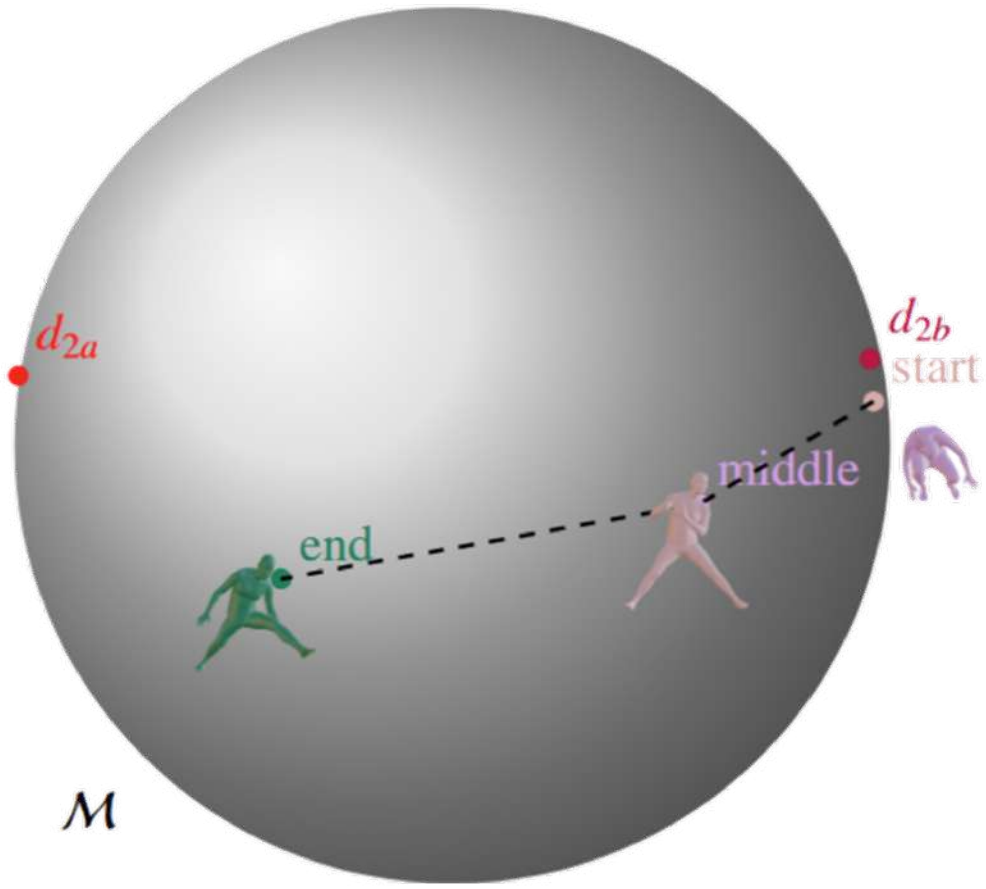


Figure 6.1 Visualization of SIDER2 interpolation between three poses on the manifold $\mathcal{M}$. First, the control points d_{2a} and d_{2b} are computed via geodesic extrapolation from the start, middle, and end poses, respectively. These control points define the curve and are used to compute the inner and outer interpolation points. This construction ensures that the resulting curve (dashed line) passes exactly through the intermediate pose, a property crucial for our multi-keyframe bundle-solving formulation.

An illustration is shown in Figure 6.1. Conceptually, SIDER2 extends quadratic Bézier curves to S^n by replacing linear segments with geodesics. Unlike classical Bézier (which only interpolates endpoints), SIDER2 *constrains the curve to pass through* the middle keypoint $\mathbf{q}_2$—a crucial property for latent keyframe solving (Section 6.2.5). Because SPoser's latent space is trained to respect hyperspherical geometry, these operations transfer directly to our learned manifold $\mathcal{M}$.

6.2.4 Manifold Optimization

Optimizing *directly* in latent space improves both efficiency and stability. We therefore perform constrained optimization on the hypersphere using projected gradients and retractions to remain on $\mathbb{S}^n$ throughout.

Let $\mathbf{z} \in \mathbb{S}^n$ parameterize the state and $f(\mathbf{z})$ be our objective. We compute the Euclidean gradient $\nabla f(\mathbf{z})$, then project it to the tangent space $T_{\mathbf{z}}\mathcal{M}$:

$$\mathbf{g}_{\text{tangent}} = \nabla f(\mathbf{z}) - (\nabla f(\mathbf{z}) \cdot \mathbf{z})\, \mathbf{z}.$$

A step in $T_{\mathbf{z}}\mathcal{M}$ is then taken,

$$\mathbf{z}_{\text{new}} = \mathbf{z} + \alpha\, \mathbf{g}_{\text{tangent}},$$

followed by a retraction back to the sphere,

$$\mathbf{z}_{\text{new}} = \frac{\mathbf{z}_{\text{new}}}{\|\mathbf{z}_{\text{new}}\|}.$$

This ensures feasibility at each iteration without additional penalty terms.

Require: $\mathbf{z} \in \mathcal{M}$
Ensure: $\|\mathbf{z}\| = 1$
1: **while** not converged **do**
2: Compute $\nabla f(\mathbf{z})$
3: $\mathbf{g}_{\text{tangent}} \leftarrow \nabla f(\mathbf{z}) - (\nabla f(\mathbf{z}) \cdot \mathbf{z})\, \mathbf{z}$
4: $\mathbf{z} \leftarrow \mathbf{z} + \alpha\, \mathbf{g}_{\text{tangent}}$
5: $\mathbf{z} \leftarrow \frac{\mathbf{z}}{\|\mathbf{z}\|}$
6: **end while**

Algorithm 1: Spherical Optimization with Projected Gradient Descent and Retraction

6.2.5 Latent Keyframe Bundle Solving

A standard single-frame estimate at time t solves

$$\underset{\mathbf{z}^t, \boldsymbol{\beta}^t, \mathbf{T}^t}{\operatorname{argmin}} \; \mathcal{E}^t_{data} + \mathcal{E}^t_{prior}, \tag{6.5}$$

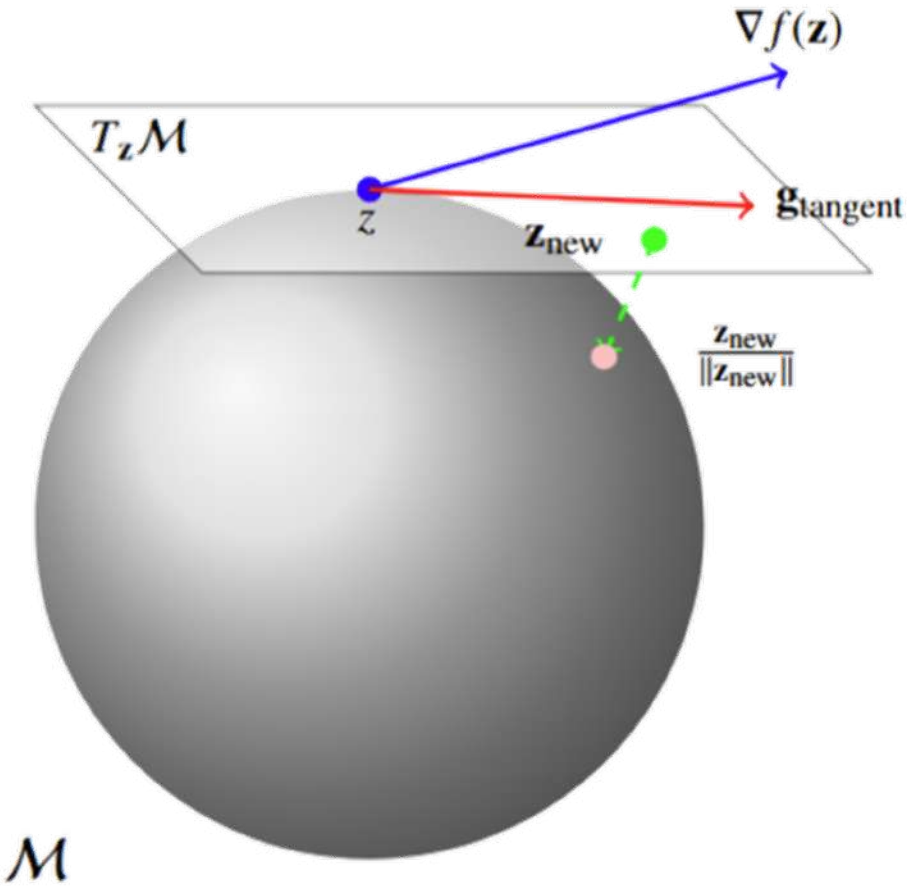


Figure 6.2 Illustration of the proposed optimization scheme on the learned manifold $\mathcal{M}$. After computing the gradient $\nabla f(\mathbf{z})$, it is projected onto the tangent space $T_{\mathbf{z}}\mathcal{M}$ to obtain $\mathbf{g}_{\text{tangent}}$, ensuring that the update direction remains tangential to the manifold. A new point $\mathbf{z}_{\text{new}}$ is then computed by updating within this tangent space, followed by a retraction step that maps $\mathbf{z}_{\text{new}}$ back onto the manifold, preserving the hyperspherical constraint throughout the optimization.

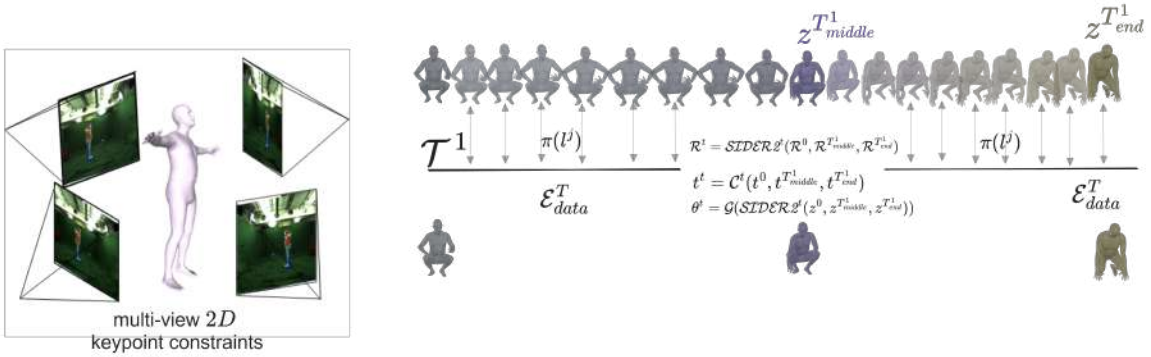


Figure 6.3 Overview of the BundleMoCap++ pipeline. BundleMoCap++ fits an articulated template mesh to 2D keypoint observations from a sparse set of multi-view videos. Instead of iteratively optimizing pose parameters for each frame, it optimizes two latent codes, $\mathbf{z}^{t_{middle}}$ and $\mathbf{z}^{t_{end}}$, corresponding to the pose parameters $\boldsymbol{\theta}^{t_{middle}} = \mathcal{G}(\mathbf{z}^{t_{middle}})$ and $\boldsymbol{\theta}^{t_{end}} = \mathcal{G}(\mathbf{z}^{t_{end}})$ for the middle and end keyframes, respectively. Intermediate poses, root orientations, and translations are reconstructed via interpolation, visually represented by the blending between the start, middle, and end keyframes. A sliding-window optimization strategy is employed, where only the first frame is fitted independently; each subsequent temporal window $\mathcal{T}^i$ optimizes only the next two latent keyframes (t^i_{middle}, t^i_{end}), while the intermediate frames are reconstructed using the previously optimized keyframe (t^{i-1}_{end} as t^i_{start}). All reconstructed frames are jointly constrained by the corresponding multi-view keypoint observations through $\mathcal{E}^{\mathcal{T}}_{data}$. BundleMoCap++ achieves smooth, temporally consistent motions in a single optimization stage—without requiring per-frame initialization or explicit motion smoothness objectives—while maintaining state-of-the-art accuracy.

where, unlike angle-space formulations [63], we optimize a latent code $\mathbf{z} \in \mathbb{R}^d$ that decodes to pose $\boldsymbol{\theta} = \mathcal{G}(\mathbf{z})$ via a fixed, pre-trained generator. Low-dimensional optimization plus an explicit latent prior [4, 148, 175] improves conditioning, prevents degeneration, and supports staged annealing of prior weights. However, per-frame multi-stage solves are computationally heavy, even when shape $\boldsymbol{\beta}$ is fixed.

By operating directly on $\mathcal{M}$, the latent codes remain feasible without additional regularization, letting us *drop* the prior in favor of a data-only objective. Moreover, the SIDER2 interpolation (Section 6.2.3) enables a *windowed* solve over $\mathcal{T} := [0, \ldots, T]$ using only three latent keyframes at 0, T_{middle}, and T_{end}:

$$\underset{\mathbf{z}^0, \mathbf{z}^{T_{middle}}, \mathbf{z}^{T_{end}}, \mathbf{T}^0, \mathbf{T}^T}{\operatorname{argmin}} \quad \mathcal{E}^{\mathcal{T}}_{data}, \tag{6.6}$$

with a fixed initial shape $\boldsymbol{\beta}$ and a windowed data term,

$$\mathcal{E}^{\mathcal{T}}_{data} = \sum_{t=1}^{T} \mathcal{E}^{t}_{data}, \tag{6.7}$$

where

$$\mathcal{E}^{t}_{data} = \lambda_R \sum_{cam}^{Cam} \sum_{i}^{J} c_i \, r\left(\mathbf{k}^t_i - \mathbf{k}^t_{\mathrm{det},i}\right), \tag{6.8}$$

where Cam defines the number of cameras, c_i the confidence per estimated keypoint, $\mathbf{k}_i$ the regressed keypoint using $\mathcal{J}$ function, and $\mathbf{k}_{det,i}$ the estimated keypoint from the AI model. We adopt the Geman–McClure penalty r in place of a simple mean squared error to better handle outliers, weighted by λ_R.

Intermediate frames across the window are reconstructed by SIDER2:

$$\boldsymbol{\theta}^t = \mathcal{G}\Big(\mathrm{SIDER2}^t(\mathbf{z}^0, \mathbf{z}^{T_{middle}}, \mathbf{z}^{T_{end}})\Big), \tag{6.9}$$

$$\begin{bmatrix} \mathbf{R}^t & \mathbf{t}^t \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathrm{SIDER2}^t\left(\mathbf{R}^0, \mathbf{R}^{T_{middle}}, \mathbf{R}^{T_{end}}\right) & \mathcal{C}^t\left(\mathbf{t}^0, \mathbf{t}^{T_{middle}}, \mathbf{t}^{T_{end}}\right) \\ \mathbf{0} & 1 \end{bmatrix}, \tag{6.10}$$

where $\mathrm{SIDER2}^t$ and $\mathcal{C}^t$ denote SIDER2 on $\mathbb{SO}(3)^P$ (for $n = 3$) and a cubic interpolation for translations, each mapping t to $[0, 1]$ within $\mathcal{T}$.

Critically, this relies on a generator $\mathcal{G}$ that learns an expressive, smooth manifold $\mathcal{M}$ so that SIDER2 yields realistic pose trajectories. The hyperspherical constraint ensures geometric consistency, allowing spherical interpolation techniques to be applied cleanly.

6.3 Experiments

6.3.1 Implementation details

We employ SMPL [1] as the body model $\mathcal{B}$, optimizing only the body-joint degrees of freedom and ignoring expressive parts (hands and face). Our **SPoser** is trained following [4], yielding a well-conditioned decoder $\mathcal{G}$ over the latent manifold $\mathcal{M}$ of plausible human poses. 2D keypoint constraints $\mathbf{k}_{det}$ and per-joint confidences c_i are obtained with OpenPose [69]. Optimization follows Algorithm 1 and Section 6.2.4, using a strong Wolfe line-search for step sizes. We cap each solve at 30 iterations to balance speed and accuracy. All experiments are implemented in a custom PyTorch framework [198, 199] (using version *1.12*). We set the reprojection data weight to 1.0, run a single-stage solve, and use a temporal window of length $|\mathcal{T}| = 20$. Hyperspherical operations are implemented with Geomstats [200].

6.3.2 Sliding Window Optimization

Our formulation admits an efficient sliding-window implementation. Rather than solving (6.6) for *three* latent keyframes every time, we fix the first keyframe $(\mathbf{z}^0, \mathbf{T}^0)$ and optimize only the next two $(\mathbf{z}^{T_{middle}}, \mathbf{T}^{T_{end}})$. We initialize the first window $\mathcal{T}^1$ by a single-frame fit at t=0, seeded by an HMR-style regressor [71], which also initializes the shape $\boldsymbol{\beta}$ (kept fixed thereafter). The window then slides: the end keyframe of $\mathcal{T}^1$ becomes the start keyframe of $\mathcal{T}^2$, which we hold fixed while optimizing the next pair, and so on. Thus, an F-frame sequence is partitioned into $N = F/T$ windows $\{\mathcal{T}^i\}_{i=1}^N$ of size $|\mathcal{T}^i| = T{+}1$ and represented by $N{+}1$ keyframes $(\mathbf{z}^i, \mathbf{T}^i)$. For fairness, we use the same initialization protocol (HMR seed, fixed $\boldsymbol{\beta}$) for all compared methods.

6.3.3 Datasets

We evaluate on two standard multi-view benchmarks spanning a broad range of motion difficulty.

Human3.6M [201]: 3.6M frames from 4 synchronized cameras, with 3D joints from marker-based MoCap; 11 subjects (S1, S5, S6, S7, S8 train; S9, S11 test following [6]).

MPI-INF-3DHP [202]: multi-view, markerless 3D pose data. We follow prior work and use only the training split's multi-view subset (14 views), testing on subject S8.

We use views $\{0, 2, 7, 8\}$ for our experiments.

6.3.4 Metrics

MoCap Metrics. We report MPJPE (mm), joint-level RMSE, angular error (MAE) of the kinematic chain, and PA-MPJPE for Procrustes-aligned comparisons with monocular methods. Accuracy is also measured via PCK at 3 and 7 cm.

Temporal Smoothness. We compute acceleration error as in [15, 121], averaging the difference between predicted and ground-truth joint accelerations (mm/s^2). Following [203], we further plot a kinematic angle (knee flexion: hip–knee–ankle) to visualize jitter; spiky curves indicate instability.

Foot Skating (FS). As in [204], we measure foot-skate (cm/frame) over N frames:

$$FS = \sum_{p=1}^{N} \left[v_p \left(2 - 2^{\frac{h_p}{H_{thr}}} \right) \mathbf{1}_{h_p \leq H_{thr}} \right],$$

where h_p is the height of the right toe vertex, v_p its corresponding velocity, and H_{thr}=2.5 cm.

Generation Metrics. Following [186], we evaluate realism with FID and variety with DIV. Given features extracted from 1,052 real and generated poses (THuman 2.0 reference), FID is

$$FID = \|\mu_{real} - \mu_{gen}\|^2 + \mathrm{Tr}\left(\Sigma_{real} + \Sigma_{gen} - 2\sqrt{\Sigma_{real}\Sigma_{gen}} \right), \tag{6.11}$$

with (μ, Σ) the means/covariances of the respective feature sets. For diversity, we re-encode 1,052 generated poses, split into two groups of 526, and compute

$$DIV = \frac{1}{N} \sum_{i=1}^{N} \|v_i - \tilde{v}_i\|, \quad N = 526. \tag{6.12}$$

6.3.5 Methods

As a single-frame multi-view baseline, we adopt a multi-stage fitting pipeline [5] (MuVS), but use VPoser [4] instead of a GMM prior and initialize with HMR [71].

We then compare against bundle solvers: *DCT* [5] (last-stage low-frequency basis with smoothness), *ETC* [6] (full-sequence solve with joint smoothness, adapted from monocular to multi-view), *DMMR* [8] (temporal VPoser prior; in our setup camera extrinsics are fixed, so camera solving is disabled), and *SLAHMR* [9] (VPoser + HuMoR temporal prior; we skip camera-motion solving, using known extrinsics). Finally, we include our earlier BundleMoCap variant [10] using VPoser + SLERP.

	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
MuVS	53.36 *mm*	58.54 *mm*	10.23 °	28.02%	79.57%	10.99 mm/s^2
DCT [5]	50.88 *mm*	56.72 *mm*	12.32 °	29.21%	80.41%	09.19 mm/s^2
DMMR [8]	60.69 *mm*	65.16 *mm*	11.48 °	20.93%	69.48%	09.57 mm/s^2
SLAHMR [9]	50.49*mm*	54.52 *mm*	08.57 °	28.92%	79.20%	09.02 mm/s^2
ETC [6]	72.74 *mm*	77.83 *mm*	05.73°	32.63%	84.42%	07.92 mm/s^2
BundleMoCap	38.36 *mm*	43.10 *mm*	04.31°	33.70%	86.24%	06.18mm/s^2
BundleMoCap++	**36.32***mm*	**40.93***mm*	**04.11°**	**46.27%**	**93.77%**	**02.52***mm*/s^2

Table 6.1: **Quantitative comparison with state-of-the-art methods on the Human3.6M dataset.** The table reports the average error and accuracy across all actions. **Bold red** indicates the best-performing result, orange the second-best, and yellow the third-best. Arrows beside each metric denote the direction of better performance (↑ for higher is better, ↓ for lower is better).

	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
MuVS	64.99 *mm*	76.12 *mm*	6.28°	28.20%	73.75%	14.45 mm/s^2
DCT [5]	62.43 *mm*	68.13 *mm*	6.18°	35.84%	83.77%	12.01 mm/s^2
DMMR [8]	57.51 *mm*	67.66 *mm*	6.06°	37.52%	81.11%	11.52 mm/s^2
SLAHMR [9]	61.8 *mm*	62.55 *mm*	5.74°	40.86%	83.97%	11.34 mm/s^2
ETC [6]	59.51 *mm*	61.32 *mm*	5.64°	39.30%	84.50%	10.01 mm/s^2
BundleMoCap	56.41 *mm*	59.12 *mm*	5.43°	44.51%	85.71%	07.39 mm/s^2
BundleMoCap ++	**48.27** *mm*	**56.44** *mm*	**4.33°**	**48.75%**	**89.34%**	**05.66** mm/s^2

Table 6.2: **Quantitative comparison with state-of-the-art methods on the MPI-INF-3DHP dataset.** The table reports the average error and accuracy across all actions. **Bold red** indicates the best-performing result, orange the second-best, and yellow the third-best. Arrows beside each metric denote the direction of better performance (↑ for higher is better, ↓ for lower is better).

	walking						eating						smoking					
	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
MuVS	52.45 *mm*	57.80 *mm*	23.57 °	28.73%	86.88%	10.99 mm/s^2	43.83 *mm*	48.33 *mm*	04.56 °	40.59%	93.75%	10.75 mm/s^2	41.69 *mm*	46.47*mm*	04.71°	30.49%	83.14%	10.92 mm/s^2
DCT	57.66 *mm*	55.37 *mm*	20.87 °	26.12%	90.41%	09.19 mm/s^2	43.87 *mm*	48.39 *mm*	04.57 °	40.84%	93.77%	10.94 mm/s^2	40.32*mm*	48.84 *mm*	13.64 °	33.00%	85.79%	08.99 mm/s^2
DMMR	49.21 *mm*	46.28*mm*	17.15 °	26.88%	91.34%	09.57 mm/s^2	74.98 *mm*	84.99 *mm*	06.47 °	11.68%	55.14%	12.32 mm/s^2	51.83 *mm*	61.88 *mm*	21.37 °	37.82%	82.38%	07.88 mm/s^2
SLAHMR	57.91 *mm*	55.70 *mm*	04.79°	32.86%	89.37%	09.02 mm/s^2	43.40 *mm*	48.29 *mm*	04.17°	40.33%	93.25%	09.67 mm/s^2	52.44 *mm*	55.52 *mm*	16.19 °	35.14%	87.07%	09.92 mm/s^2
ETC	42.85*mm*	48.66 *mm*	05.69 °	33.13%	90.69%	07.92 mm/s^2	37.49*mm*	41.09*mm*	04.63 °	40.30%	94.96%	06.55 mm/s^2	40.69*mm*	45.91*mm*	07.62 °	37.51%	90.04%	06.53mm/s^2
BundleMoCap	41.04*mm*	45.65*mm*	04.79°	34.38%	91.84%	06.18mm/s^2	36.48*mm*	40.34*mm*	03.96°	41.16%	95.72%	03.55mm/s^2	39.47*mm*	44.54*mm*	04.39°	38.53%	91.96%	04.47mm/s^2
BundleMoCap++	**31.92***mm*	**35.93***mm*	**04.23°**	**52.28%**	**96.90%**	**02.75***mm*/s^2	**33.16***mm*	**37.55***mm*	**03.65°**	**50.46%**	**97.01%**	**02.14***mm*/s^2	**35.74***mm*	**40.84***mm*	**03.15°**	**48.39%**	**92.89%**	**02.17***mm*/s^2
	directions						greeting						phoning					
	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
MuVS	42.66 *mm*	47.40 *mm*	04.43 °	31.52%	92.15%	11.45 mm/s^2	55.61 *mm*	52.19 *mm*	07.24°	33.41%	89.76%	10.87 mm/s^2	117.13 *mm*	136.34 *mm*	10.89 °	21.70%	61.10%	14.55 mm/s^2
DCT	42.00 *mm*	46.56 *mm*	04.41 °	32.34%	93.11%	08.32 mm/s^2	45.80 *mm*	50.97*mm*	11.26 °	29.28%	92.89%	09.75 mm/s^2	114.78 *mm*	133.55 *mm*	10.74 °	21.99%	62.03%	13.45 mm/s^2
DMMR	64.47 *mm*	72.99 *mm*	06.92 °	08.50%	69.53%	12.34 mm/s^2	55.73 *mm*	53.48 *mm*	08.12 °	29.77%	88.70%	10.28 mm/s^2	89.74*mm*	100.77*mm*	08.68°	07.32%	44.35%	12.29 mm/s^2
SLAHMR	43.12 *mm*	00.00 *mm*	00.00 °	00.00%	00.00%	10.87 mm/s^2	48.17 *mm*	58.92 *mm*	07.54 °	35.72%	92.05%	09.18 mm/s^2	114.14 *mm*	133.08 *mm*	10.23 °	21.65%	61.83%	14.18 mm/s^2
ETC	38.69*mm*	42.56*mm*	04.30°	**40.56%**	**97.74%**	05.65mm/s^2	43.61*mm*	59.23 *mm*	16.17 °	29.00%	85.69%	08.46mm/s^2	374.72 *mm*	381.96 *mm*	09.52 °	27.13%	84.30%	15.73 mm/s^2
BundleMoCap	39.24*mm*	43.14*mm*	04.37°	35.89%	95.17%	03.45mm/s^2	38.79*mm*	42.75*mm*	04.48°	37.87%	94.57%	03.25mm/s^2	42.67*mm*	50.98*mm*	04.37°	32.23%	92.61%	04.12mm/s^2
BundleMoCap++	**38.27***mm*	**42.46***mm*	**03.85°**	38.82%	96.18%	**02.75***mm*/s^2	**36.45***mm*	**39.28***mm*	**04.10°**	**39.17%**	**95.28%**	**02.89***mm*/s^2	**41.01***mm*	**46.24***mm*	**04.13°**	**44.23%**	**92.76%**	**02.44***mm*/s^2
	purchases						sitting						sittingdown					
	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
MuVS	43.89*mm*	52.56*mm*	05.38 °	35.34%	81.09%	10.02 mm/s^2	59.06 *mm*	74.45 *mm*	06.71 °	25.50%	84.65%	10.20 mm/s^2	79.03 *mm*	95.35 *mm*	09.59 °	20.47%	66.40%	12.45 mm/s^2
DCT	58.97 *mm*	60.66 *mm*	20.16 °	34.46%	90.12%	08.87 mm/s^2	53.23 *mm*	70.24 *mm*	19.38 °	30.57%	76.00%	09.91 mm/s^2	76.78*mm*	93.09*mm*	09.15 °	20.41%	65.86%	10.91 mm/s^2
DMMR	46.50 *mm*	59.17 *mm*	18.62 °	30.23%	89.43%	09.91 mm/s^2	56.35 *mm*	60.31 *mm*	20.10 °	25.33%	83.12%	09.72 mm/s^2	129.54 *mm*	147.91 *mm*	15.59 °	03.10%	23.00%	09.91 mm/s^2
SLAHMR	57.50 *mm*	56.55 *mm*	07.24 °	31.10%	80.84%	07.29 mm/s^2	50.73*mm*	60.20*mm*	16.11 °	27.32%	82.26%	08.39 mm/s^2	78.46 *mm*	95.11 *mm*	09.07 °	19.62%	65.80%	08.19 mm/s^2
ETC	59.81 *mm*	64.75 *mm*	04.89°	**37.53%**	90.55%	06.36 mm/s^2	102.05 *mm*	110.21 *mm*	05.80°	31.85%	83.44%	10.02 mm/s^2	105.07 *mm*	112.22 *mm*	07.14°	28.16%	80.80%	5.29mm/s^2
BundleMoCap	41.69*mm*	46.21*mm*	04.33°	36.24%	90.62%	04.25mm/s^2	46.40*mm*	55.23*mm*	05.02°	33.45%	85.30%	03.10mm/s^2	49.19*mm*	56.51*mm*	**06.60°**	27.91%	**81.35%**	04.27mm/s^2
BundleMoCap	**35.94***mm*	**39.58***mm*	**04.13°**	**45.12%**	**93.21%**	**03.20***mm*/s^2	**43.29***mm*	**50.83***mm*	**04.87°**	**37.74%**	**87.37%**	**02.85***mm*/s^2	**48.73***mm*	**55.27***mm*	06.79°	**30.01%**	81.28%	**03.70***mm*/s^2
	waiting						walkingdog						walkingtogether					
	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
MuVS	44.55*mm*	55.65 *mm*	19.55 °	28.37%	90.96%	12.09 mm/s^2	59.38 *mm*	60.77 *mm*	07.65 °	26.15%	86.68%	10.20 mm/s^2	56.83 *mm*	48.53 *mm*	07.42 °	34.33%	85.12%	12.02 mm/s^2
DCT	58.48 *mm*	48.02*mm*	20.25 °	34.20%	90.85%	10.93 mm/s^2	55.34 *mm*	48.75 *mm*	10.12 °	27.72%	86.86%	09.90 mm/s^2	42.11*mm*	50.99 *mm*	16.25 °	30.09%	91.42%	11.00 mm/s^2
DMMR	52.34 *mm*	58.14 *mm*	12.49 °	31.71%	83.50%	10.98 mm/s^2	54.50 *mm*	46.93*mm*	19.90 °	33.63%	88.27%	09.11 mm/s^2	48.54 *mm*	47.78 *mm*	06.92°	27.31%	91.00%	11.02 mm/s^2
SLAHMR	47.37 *mm*	48.23 *mm*	23.86 °	28.87%	84.82%	09.28mm/s^2	45.34 *mm*	48.34 *mm*	08.04 °	28.20%	86.37%	08.99 mm/s^2	43.35 *mm*	43.80*mm*	11.73 °	29.22%	83.91%	10.03 mm/s^2
ETC	93.03 *mm*	99.56 *mm*	05.49°	37.08%	89.19%	12.30 mm/s^2	41.40*mm*	48.47 *mm*	06.07°	31.15%	86.51%	05.35mm/s^2	42.60 *mm*	44.24 *mm*	08.83 °	34.49%	91.93%	06.47mm/s^2
BundleMoCap	41.12*mm*	46.44*mm*	05.07°	37.92%	91.27%	03.65mm/s^2	41.04*mm*	45.65*mm*	04.79°	34.38%	91.84%	06.02 mm/s^2	38.76*mm*	42.79*mm*	04.46°	36.24%	93.50%	06.34 mm/s^2
BundleMoCap++	**36.18***mm*	**41.47***mm*	**04.15°**	**51.25%**	**93.27%**	**01.88***mm*/s^2	**34.91***mm*	**39.88***mm*	**03.78°**	**49.36%**	**94.36%**	**02.78***mm*/s^2	**29.74***mm*	**33.66***mm*	**04.44°**	**57.91%**	**98.18%**	**02.27***mm*/s^2
	discussion						posing						takingphoto					
	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
MuVS	43.67 *mm*	45.88 *mm*	04.29 °	31.12%	92.80%	11.04 mm/s^2	49.47 *mm*	43.05 *mm*	21.86 °	38.66%	91.53%	08.57 mm/s^2	51.20 *mm*	56.15 *mm*	04.90 °	27.45%	86.07%	09.97 mm/s^2
DCT	44.00 *mm*	46.33 *mm*	04.09 °	33.81%	93.18%	12.02 mm/s^2	39.16*mm*	56.85 *mm*	16.28 °	37.68%	94.36%	07.45 mm/s^2	41.26*mm*	45.68*mm*	04.54 °	32.23%	86.95%	08.18 mm/s^2
DMMR	64.68 *mm*	73.45 *mm*	06.07 °	08.75%	70.53%	12.94 mm/s^2	42.78 *mm*	45.86 *mm*	17.10 °	44.29%	91.45%	09.27 mm/s^2	88.37 *mm*	99.68 *mm*	07.67 °	11.69%	48.73%	10.44 mm/s^2
SLAHMR	43.72 *mm*	45.92 *mm*	03.36°	32.97%	92.99%	10.97 mm/s^2	39.52 *mm*	40.92*mm*	10.33 °	42.83%	87.91%	09.25 mm/s^2	44.83 *mm*	50.50 *mm*	12.55 °	31.32%	86.88%	08.47 mm/s^2
ETC	39.69*mm*	42.92*mm*	04.23 °	**40.82%**	**97.75%**	07.43mm/s^2	51.11 *mm*	56.21 *mm*	04.71°	39.45%	93.97%	05.41 mm/s^2	55.80 *mm*	60.76 *mm*	04.13°	32.34%	90.94%	04.58 mm/s^2
BundleMoCap	39.54*mm*	42.14*mm*	04.07°	36.73%	95.57%	03.48mm/s^2	**36.06***mm*	**40.00***mm*	04.08°	44.29%	**96.49%**	03.35mm/s^2	42.29*mm*	47.14*mm*	04.19°	31.99%	92.08%	03.94mm/s^2
BundleMoCap++	**37.23***mm*	**41.50***mm*	**03.33°**	38.84%	97.19%	**02.43***mm*/s^2	**26.51***mm*	**30.07***mm*	**03.23°**	**67.23%**	**99.07%**	**01.73***mm*/s^2	**35.67***mm*	**39.45***mm*	**03.82°**	**43.27%**	**95.66%**	**01.75***mm*/s^2

Table 6.3: Quantitative comparison against other methods on the Human3.6M dataset, per action. **Bold red** marks the best performing row, orange the second best, and yellow the third. The arrows next to each metric denote the direction of better performance. BundleMoCap++ is the best-performing method across most of the actions and metrics. It is worth noting that in actions like *sitting*, the gain is more evident as 2D keypoint detections are more likely to fail due to heavy occlusions. This makes the influence of the longer temporal window more evident in the results

6.3.6 Analysis

Is the method robust to outliers? Across Human3.6M and MPI-INF-3DHP (Tables 6.1 to 6.3), **BundleMoCap++** outperforms multi-stage, smoothness-regularized methods. Gains are largest on difficult actions with occlusions and keypoint inversions. Our hyperspherical manifold plus high-order interpolation recovers smooth transitions without autoregressive priors (HuMoR in SLAHMR [9, 99]) or recurrent latent smoothing (DMMR [8]). Competing bundle solvers optimize per-frame latents jointly; we solve only two latent keyframes and reconstruct intermediates via SIDER2, which avoids converging to spurious poses caused by conflicting multi-view detections. Figure 6.8 highlights sequences with severe keypoint outliers where other methods become temporally inconsistent, whereas BundleMoCap++ remains coherent.

Are the generated motions natural? Despite noisy detections, BundleMoCap++ yields smooth trajectories *without* an explicit temporal smoothness loss. We visualize knee-flexion angles (hip–knee–ankle): our curve is notably less jittery than a segment solver [5] and a full-sequence joint-smoothness solver [6] (Fig. 6.4). Supplementary videos (same color coding as Fig. 6.8) further illustrate natural motion. Because our reconstruction respects both the pose prior and the 2D constraints, realism is preserved; the prior weighs in more when detection confidence is low, acting as an infilling prior.

	FS ↓
$MuVS$	0.09 cm/f
DCT	0.09 cm/f
DMMR	0.06 cm/f
SLAHMR	0.07 cm/f
ETC	0.06 cm/f
BundleMoCap	0.07 cm/f
BundleMoCap++	**0.03** cm/f

Table 6.4: **Foot-skating comparison across baselines on the Walking actions of the Human3.6M dataset.** The metric is reported only for movement-based actions, as including static actions (e.g., sitting), would introduce bias into the evaluation.

As corroborated by the foot-skating metric [204] in Table 6.4, our method exhibits less skating than baselines. While interpolation methods can introduce "smooth skating," BundleMoCap++ avoids this—likely due to the extra keyframe, which better preserves high-frequency motion components.

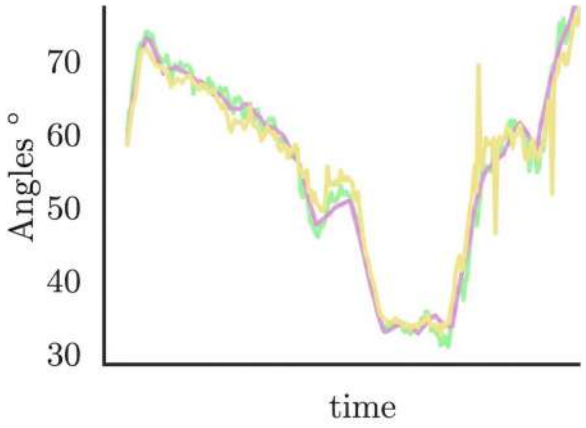


Figure 6.4 Knee flexion angle segment for the *Sitting Down* action (subject S9, Human3.6M dataset). Although BundleMoCap++ does not explicitly enforce temporal consistency during optimization, it produces smooth motion comparable to state-of-the-art methods such as DCT [5] (green) and ETC [6] (yellow), both of which include explicit smoothness objectives. This smoothness emerges naturally from the expressiveness of the learned pose manifold $\mathcal{M}$, which ensures locally continuous transitions across poses and enables fluid motion capture. Importantly, this implicit smoothness does not compromise accuracy, as supported by the quantitative results in Tables 6.3 and 6.2.

Is the proposed method efficient? BundleMoCap++ scales favorably with sequence length. Compared to our earlier BundleMoCap (single first-frame solve, then 10-frame bundles with a single latent), BundleMoCap++ uses two keyframes over 20-frame bundles, improving both accuracy and runtime. Other pipelines require costly passes—per-frame initialization (ETC [6]), full single-frame passes before bundling (DCT/SLAHMR [5, 9]), or multi-stage per-frame latent optimization (DMMR [8]). Like [10], we solve the entire sequence in a *single*, sliding-window stage. Indicative runtimes on a 4-view, 2000-frame sequence: **BundleMoCap++** $\approx$ 45 min; BundleMoCap $\approx$ 60 min; DMMR $\approx$ 24 h (4 stages); ETC $\approx$ 3 h; DCT $\approx$ 40 min init + 9 h optimize; SLAHMR $\approx$ 6 h. Even excluding detector cost and potential engineering improvements, the relative complexity gap is clear (Fig. 6.5).

Are errors propagated during optimization? Sliding windows can drift, but BundleMoCap++ mitigates this by constraining an entire 20-frame window rather than only two keyframes. This reduces error carry-over between windows. Quantitatively (Table 6.2), long sequences spanning minutes do not exhibit worsening metrics, indicating limited drift. A remaining assumption is that short geodesic paths on $\mathcal{M}$ reconstruct short motions faithfully; future work will study larger windows/more

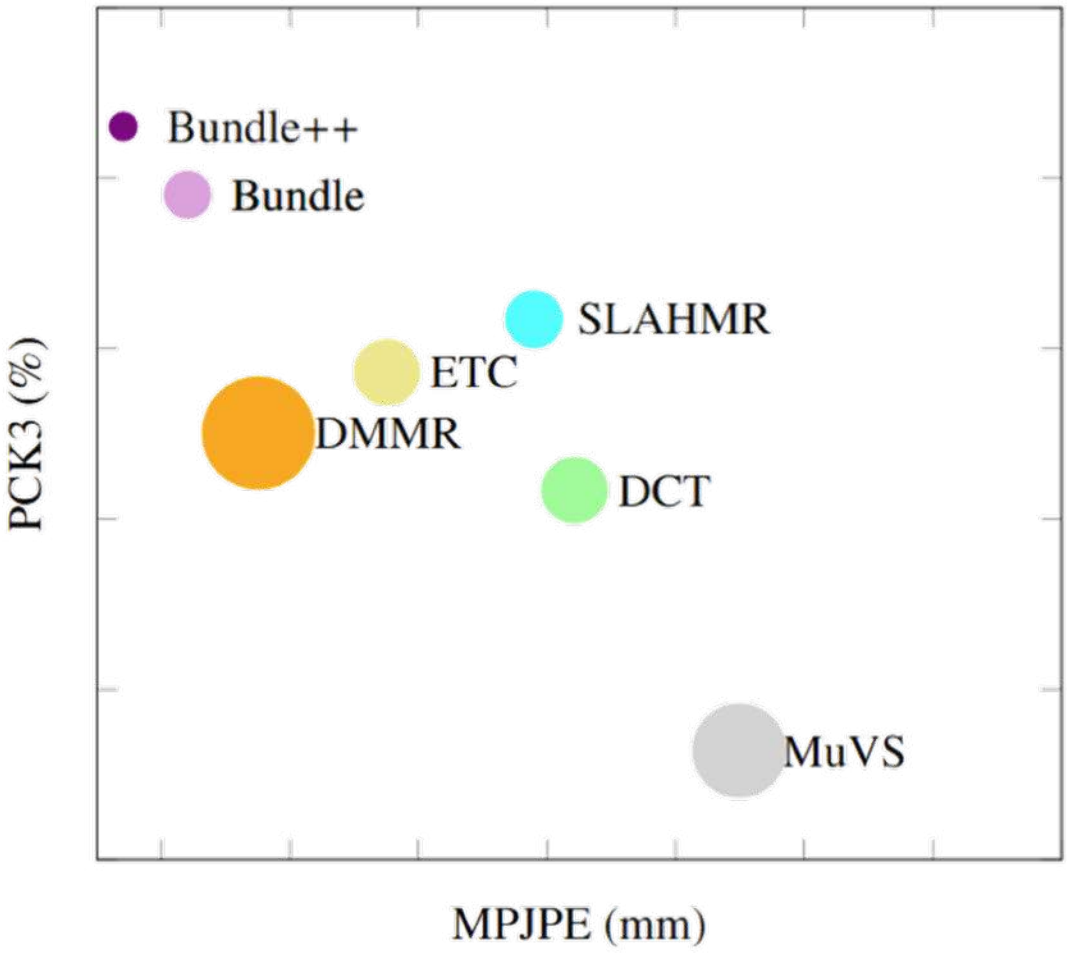


Figure 6.5 Performance–efficiency trade-off across different methods. The horizontal and vertical axes represent performance metrics, while point size indicates runtime efficiency. BundleMoCap++ achieves competitive accuracy with minimal computational cost, requiring neither 3D initialization nor explicit smoothness objectives. Its single-stage design substantially improves efficiency, making it well-suited for practical real-time applications. Furthermore, direct optimization on the manifold $\mathcal{M}$ promotes faster convergence and higher accuracy.

keyframes vs. expressivity and drift.

Which is the most appropriate human pose prior? A well-structured latent manifold improves interpolation diversity, realism, and even clustering. We compare **SPoser** (explicit hypersphere) against SVAE [138] and an SO(3)-homeomorphism variant [140], as well as VPoser [4] and RVPoser [7]. SPoser achieves the best overall

	Synthesis		Reconstruction		
	FID ↓	DIV ↑	MPJPE ↓	PCK3 ↑	PCK7 ↑
VPoser	9.94	12.11	26.21mm	62.33%	89.82%
RVPoser	8.57	13.24	24.91mm	**70.01%**	**94.05%**
LieVAE	14.55	13.20	32.45 mm	55.32%	85.65%
SVAE	12.22	**19.51**	27.81 mm	66.14%	91.07%
SPoser(ours)	**3.45**	18.12	**24.88** mm	69.67%	93.99%

Table 6.5: **Comparison of different VAE architectures.** SPoser achieves a balanced trade-off between reconstruction fidelity and generative quality, making it particularly suitable for multi-keyframe bundle solving where both properties are essential. In scenarios with missing or low-confidence observations, the model's generative capacity enables realistic pose synthesis, whereas its strong reconstruction ability ensures accurate fitting when high-confidence observations are available during optimization.

trade-off—maintaining accuracy without sacrificing diversity (where SVAE tends to trade one for the other). UMAP plots (Figure 4.9) visualize broader, more uniform coverage of the pose manifold.

Figure 6.6 UMAP projections of "real" and generated pose samples. The plots visualize ground-truth samples alongside synthetic poses produced by SPoser (left), RVPoser [7] (middle), and VPoser [4] (right). SPoser uniquely maps human poses onto a hyperspherical manifold, enabling the use of advanced spherical interpolation schemes while achieving broader and more uniform coverage of the ground-truth pose distribution.

Which interpolation technique is most effective? **SIDER2** is especially suitable for multi-keyframe solving: by construction it passes *through* the middle keyframe (unlike Bézier in $\mathbb{R}^n$) and composes SLERPs to achieve C^n continuity for $n \geq 2$, mak-

ing it extensible to more keyframes. SLERP performs well but is inherently two-point, which limits long-window fidelity. Figure 6.7 shows SIDER generating more diverse and plausible trajectories.

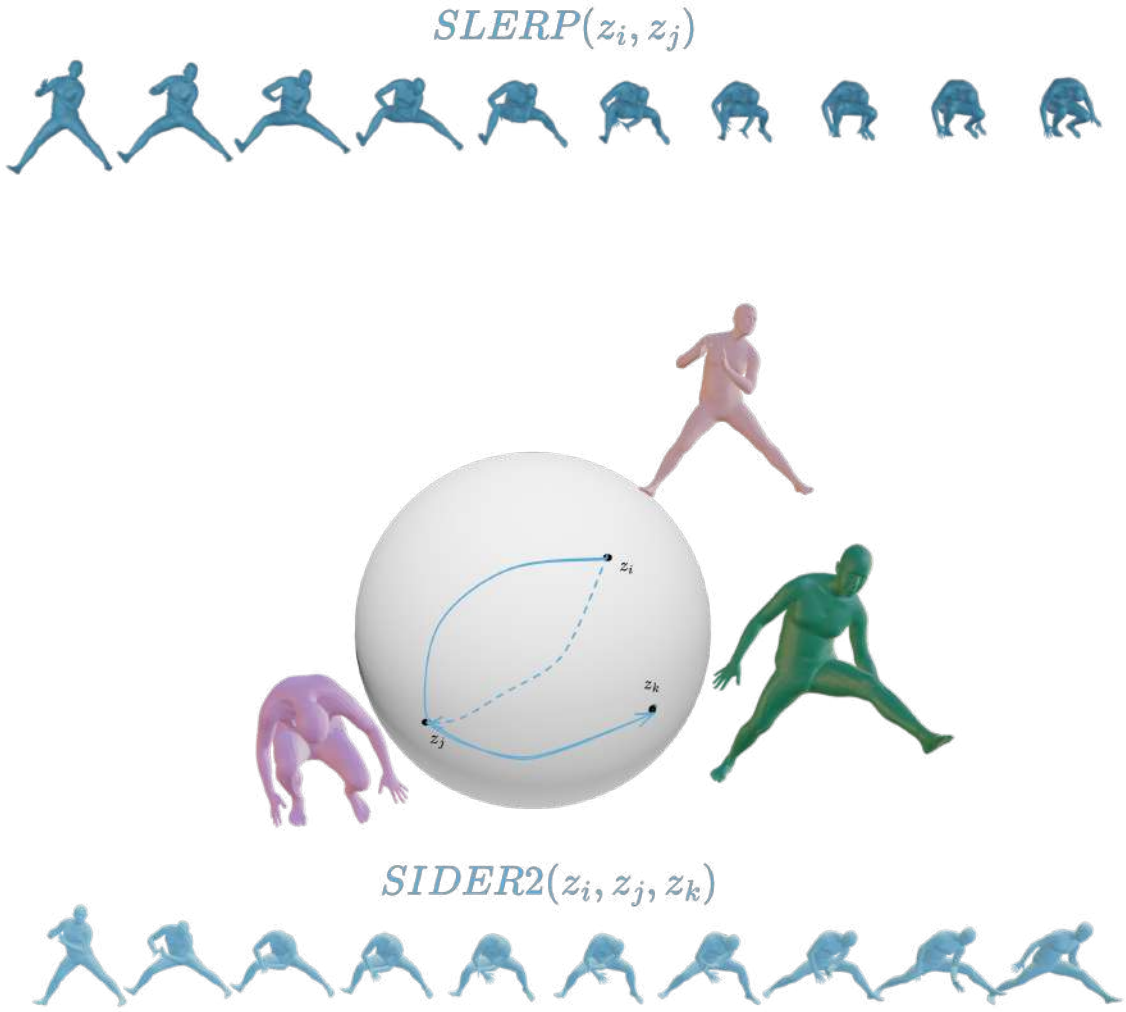


Figure 6.7 Comparison of SLERP and SIDER2 interpolation. SIDER2 enables interpolation among three latent vectors simultaneously, whereas SLERP is limited to two. This capability allows SIDER2 to generate more diverse and realistic intermediate samples between latent anchor poses, making it better suited for our multi-keyframe solving framework. Moreover, SIDER2 supports optimization over longer temporal windows without compromising performance, while improving both computational efficiency and robustness to outliers.

Effect of Direct Optimization. Direct manifold optimization preserves geometry, stabilizes convergence, and yields more faithful representations. Figure 6.5 shows

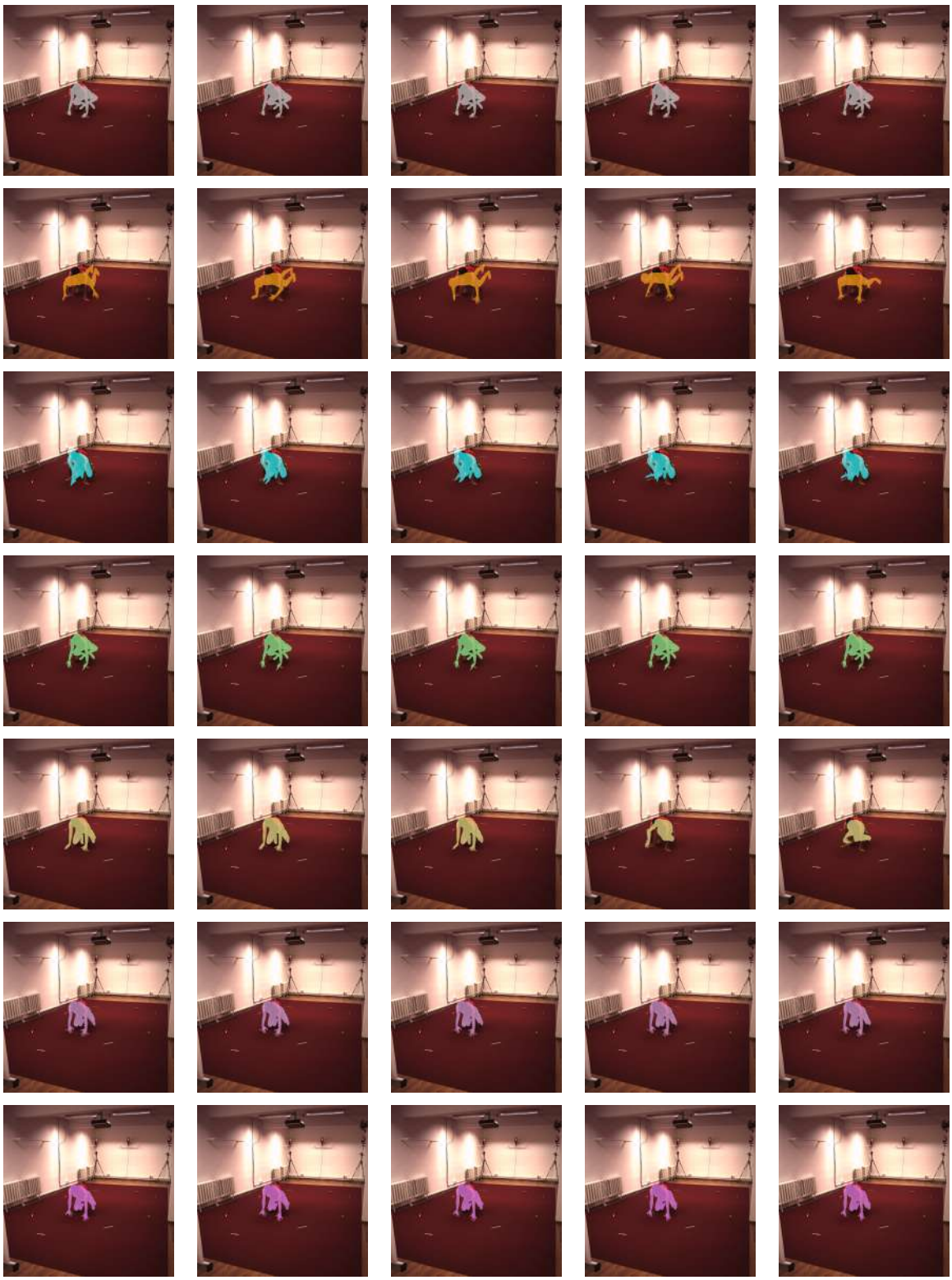


Figure 6.8 Qualitative comparison on the Human3.6M dataset for the *SittingDown* action (subject S9). Each column corresponds to a consecutive frame, while rows (top to bottom) show results from $MuVS$ (gray), DMMR [8] (orange), SLAHMR [9] (cyan), DCT [5] (green), ETC [6] (yellow), BundleMoCap [10] (magenta), and the proposed BundleMoCap++ (violet). Our method demonstrates strong robustness to occlusions and erroneous keypoint detections that significantly degrade competing approaches, particularly under sparse-view conditions. This en-

our method's favorable accuracy–efficiency balance. Because variables remain on $\mathcal{M}$ by construction, we can optimize using only the data term—no extra priors or hyper-parameters—simplifying training and tuning.

Comparison with Pose Estimators. Framewise pose estimators predict joints independently and often ignore bone-length consistency or rotations, requiring additional IK to obtain full motion. Our method produces full-body motion from 2D constraints while respecting structure. To compare directly, we evaluate against [13] on MPI-INF-3DHP (unseen for their training) under matched conditions; see Table 6.6. BundleMoCap++ achieves smoother motion without degrading accuracy.

	MPJPE ↓	RMSE ↓	PCK3 ↑	PCK7 ↑	accel ↓
MVN [13]	85.34 mm	114.03 mm	10.76%	62.48%	07.45 mm/s^2
BundleMoCap++	**48.27** mm	**56.44** mm	**48.75**%	**89.34**%	**05.34** mm/s^2

Table 6.6: **Quantitative comparison with MVN [13] on the MPI-INF-3DHP test set.** Angular error is not reported, as MVN estimates only joint positions and does not provide full rotational information.

Comparison with Monocular Methods. We also compare to [14, 15]. Because monocular reconstructions lack scale, we pelvis-align predictions to ground truth and report these aligned metrics (Table 6.7). BundleMoCap++ outperforms even PA results and produces smoother trajectories than [15], which specifically targets temporal coherence.

	MPJPE ↓	PAMPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
H4D		49.05 mm	56.70 mm	7.56°	40.34%	82.35%	18.10 mm/s^2
TCMR		110.4 mm	135.4 mm	15.56°	10.94%	55.46%	10.70 mm/s^2
BundleMoCap++	**48.27** mm		**56.44** mm	**4.33**°	**48.75**%	**89.34**%	**05.66** mm/s^2

Table 6.7: **Quantitative comparison with monocular methods H4D [14] and TCMR [15] on the MPI-INF-3DHP test set.** For a fair comparison, Procrustes-aligned (PA) metrics are reported.

Effect of Temporal Window. Table 6.8 compares SLERP-based interpolation at $\mathcal{T}$=10 vs. $\mathcal{T}$=20. SLERP is competitive for short windows but degrades at 20 frames—likely due to limited temporal expressivity—leading to oversmoothing and loss of high-frequency detail. By contrast, **SIDER** maintains performance with longer windows, capturing motion dynamics more faithfully and scaling naturally to broader contexts.

	MPJPE ↓	RMSE ↓	PCK3 ↑	PCK7 ↑	Acc ↓
$SLERP_{T=10}$	**40.05** mm	**45.52** mm	**39.10**%	**91.47**%	**03.84** mm/s^2
$SLERP_{T=20}$	41.26 mm	47.24 mm	38.20%	89.35%	03.99 mm/s^2

Table 6.8: **Quantitative comparison across different temporal window sizes $\mathcal{T}$ using $SLERP$ interpolation of $\boldsymbol{\theta}$ on the Human3.6M dataset.** Extending the temporal window when using $SLERP$ leads to degraded motion quality, highlighting its limitations for longer sequences.

Effect of high-order interpolation on global orientation. Table 6.9 compares SLERP vs. SIDER for $\mathbf{R}$. Applying high-order interpolation coherently across $\boldsymbol{\theta}$, $\mathbf{R}$, and $\mathbf{t}$ yield the best results, as additional keyframes capture fast motions and reduce oversmoothing.

	MPJPE ↓	RMSE ↓	MAE ↓	PCK3 ↑	PCK7 ↑	accel ↓
$SLERP$	37.82 mm	42.62 mm	04.15°	35.86%	86.94%	3.66 mm/s^2
$SIDER2$	**36.32** mm	**40.93** mm	**04.09°**	**46.27**%	**93.97**%	**02.52** mm/s^2

Table 6.9: **Comparison of $SLERP$ and $SIDER$ interpolation methods on a sequence from the MPI-INF-3DHP dataset.** Applying high-order interpolation across $\boldsymbol{\theta}$, $\mathbf{R}$, and $\mathbf{t}$ components results in more accurate motion reconstruction and mitigates over-smoothing effects.

Real-world experiments. Apart from rigorously evaluating the proposed method across different datasets and several baselines, we also compare it in a real-world setting with a sparse deployment of low-cost sensors and in challenging environmental conditions (e.g., outdoor capture and varying lighting). To that end, we developed a distributed real-time MoCap system using edge-AI devices—specifically the Luxonis OAK-D Pro PoE sensors to collect synchronized multi-view video streams, at 60 FPS. Then we apply spatial calibration as in [25] and the solving approach described in this chapter to the captured videos. Qualitative results depicted in Figure 6.9, as well as in the accompanying video, demonstrate robust performance in real-world conditions with challenging occlusions and viewpoints, even in a demanding sport scenario in which a subject performs dynamic jumps and headings.

The proposed method enables the extraction of high-quality MoCap data, ready for use in a range of downstream applications, from sports and biomechanics to animation and VFX.

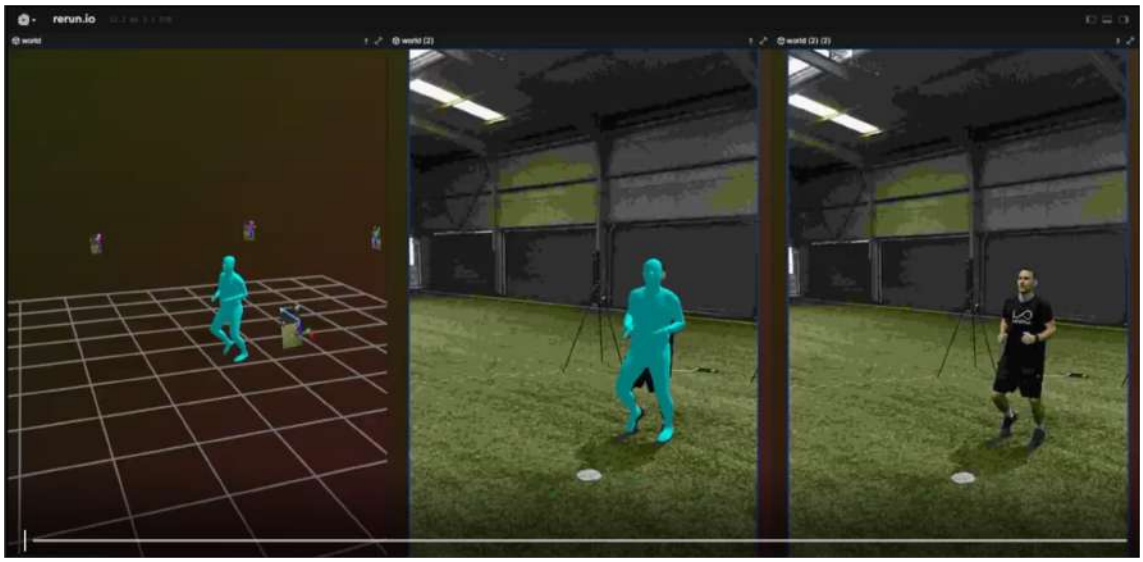


Figure 6.9 Demonstration of our markerless MoCap pipeline in a challenging football scenario. **Left:** Multi-view camera setup and reconstructed scene in 3D. **Middle:** Solved body motion overlaid on the reference camera view. **Right:** Raw reference image. The full video of this sequence is available at `https://www.youtube.com/watch?v=XTZ6jWjKtQQ&feature=youtu.be`.

6.4 Conclusions

We introduced a markerless MoCap pipeline for sparse multi-view videos that (i) learns a hyperspherical pose prior via a VAE, (ii) optimizes *directly* on the latent manifold, and (iii) reconstructs temporal bundles through manifold interpolation between three latent keyframes. This design yields an efficient solver that produces smooth, realistic motion and remains robust to outlier 2D observations.

A central finding is that *geometry-aware* interpolation can be applied reliably in the latent space of a well-structured pose prior. By respecting the hypersphere's geometry, higher-order schemes (e.g., SIDER2) enable accurate reconstruction across longer windows without explicit smoothness penalties. This opens the door to richer temporal parameterizations (more keyframes, wider windows) at modest computational cost.

Limitations and opportunities. BundleMoCap++ assumes locally smooth geodesic paths on the learned manifold; improving local fidelity with stronger generative priors could further enhance reconstruction under fast or complex motions. Conversely, the diversity of any learned prior is bounded by its training set, suggesting

benefits from broader or domain-adapted data.

Future directions. Promising extensions include:

- *Global SE(3) interpolation:* replacing independent root rotation/translation fits with screw-theory (SE(3)) interpolation to better capture coordinated root motion.
- *More keyframes & longer windows:* scaling SIDER-n to additional anchors while preserving stability and avoiding drift.
- *Multi-person and human–object scenes:* extending constraints to inter-person and contact priors for interaction-heavy sequences.
- *Monocular deployment:* applying the framework to single-view videos, where jitter and missing keypoints are more severe.
- *Hand Solving:* applying the framework for solving a parametric hand model from a sparse set of RGB cameras.

In summary, geometry-consistent latent interpolation coupled with manifold-constrained optimization provides a compact, effective route to temporally coherent motion capture in sparse-view settings, and establishes a foundation for broader scene-understanding scenarios.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this work, we have demonstrated how we can leverage **representation learning** for addressing a cascade of challenges in Motion Capture, aiming to add our small part to making Motion Capture widely available. Our contributions show that by learning a rich, underlying manifold of human poses, we can develop solutions that are more robust, efficient, and adaptive than traditional approaches, particularly for low-cost and markerless systems.

Our investigation began in the fourth chapter by leveraging **representation learning** to tackle the foundational problem of long-tailed data distribution in human datasets, which negatively affects the quality of the AI models. By using a Variational Autoencoder to learn a continuous manifold of human poses, we were able to balance existing imbalanced Motion Capture datasets and build more accurate data-driven models, able to excel in rare and out-of-distribution poses. We demonstrated the effectiveness of our approach on the challenging marker-solving task, where our models outperformed state-of-the-art models by a large margin, especially on out-of-distribution poses. Building on this, the fifth chapter addressed a critical consequence: the uncertainty inherent in these models' estimates when applied to noisy, real-world data. To alleviate that, we introduced a robust optimization framework that makes the system aware of the AI's own limitations, learning to distrust its outputs when the input data or the models' predictions are poor. Finally, our sixth chapter presented the culmination of this representation-centric philoso-

phy by tackling markerless MoCap from a sparse set of sensors. Here, we replaced complex, per-frame optimization entirely with a more powerful approach: solving for two compact **latent codes** within our learned pose manifold. The resulting motion is generated via **manifold interpolation** directly in the latent space, providing a single-stage solution that is inherently smooth and robust to outliers. The unifying theme of this Thesis is the paradigm shift from explicit, handcrafted models to powerful, learned representations. Instead of defining rigid rules for motion dynamics or noise distributions, we have shown that by focusing on learning a high-quality underlying representation of human pose, subsequent tasks like denoising, uncertainty estimation, and temporal solving become far more tractable. This approach allows the model to capture the complex, non-linear correlations of human motion in a way that traditional methods cannot.

7.2 Future Work

Though MoCap has achieved massive progress over the last decades, there is still a massive gap between what the current state-of-the-art permits and the requirements of modern applications, limiting its wider adoption. The major problem to handle is still about improving the efficiency and effectiveness of MoCap with as little extra time and manual work as possible. Here we list some promising new trends that build upon the principles of representation learning:

- **Monocular human motion capture:** Deep learning has shown promising results in monocular human pose estimation and motion capture, which is critical for accessible AR/VR applications that rely on a single device camera. However, the fundamental, ill-posed nature of inferring 3D poses from a 2D projection remains a significant hurdle. Current methods often fail to efficiently track both the camera's trajectory and the user's motion simultaneously.

  Following a similar approach to that in Chapter 6, we can significantly improve the quality of the estimated motion. This methodology could be extended to model camera poses as well. For instance, developing a similar $SO(3)$ prior before modeling the camera's orientation would enable efficient interpolation of camera poses. This would enforce temporal consistency in the camera's trajectory, thereby improving the system's overall accuracy. Another interesting direction would be to develop more powerful temporal priors and physics-informed models. Such advancements are essential for resolving the inherent depth ambiguities in monocular estimation and creating more robust and realistic motion capture systems.

- **Multi-person motion capture:** Capturing the motion of multiple interacting people is essential for creating believable social VR experiences and analyzing group behavior, or capturing athletes' performance. The challenges go beyond simply tracking multiple skeletons; they include frequent inter-person occlusions, identity association, and modeling the subtle semantics of social interaction. We have already made some preliminary tests modifying the temporal tracker to account for multiple people. Nevertheless, an interesting future work could explore methods for accurately capturing and modeling multi-person interactions by learning a "social interaction prior," which would encode implicit rules of proxemics, joint attention, and other non-verbal social behaviors.

- **Appearance modeling:** While current MoCap systems focus on capturing the geometry and motion of the human body, creating true digital twins for immersive experiences requires modeling dynamic appearance. Future work could explore methods for capturing and modeling the appearance of clothing, hair, and other accessories to create fully dynamic neural avatars. This involves exploiting recent advances in neural rendering—such as Neural Radiance Fields (NeRFs) and Gaussian Splatting—and extending them from static scenes to deforming, articulated subjects. The goal is to learn a canonical model of a person's appearance and a deformation field that maps it to any given pose. With this proper human modeling, which extends those presented in Chapter 2, additional color constraints could be incorporated into the solving process, further improving the quality of the captured motion.

- **Object and finger interaction:** Accurately capturing fine-grained hand-object interactions remains a frontier problem for robotics, training simulators, and VR. The challenges are significant, stemming from the hand's high degrees of freedom, severe occlusions during interaction, and the need to understand complex physical constraints. Future work must advance towards holistic models that jointly solve for human pose, hand articulation, object state, and the contact physics governing their interplay. We believe that powerful representation learning is key to this task. By integrating strong data priors directly into the optimization process, as presented in this Thesis, the output motion would be robust to occlusions and missing joints. Essentially, the ultimate goal is to capture complex manipulation and tool use with high fidelity and physical plausibility.

- **Disentangled latent space representations:** Designing a disentangled latent space for motion, where different body parts or semantic motion concepts can be controlled independently, is key to improving the interpretability and

utility of generative motion models, as we show while developing these methods. Such representations would allow for intuitive motion editing, style transfer, and controllable synthesis (e.g., altering a character's arm gesture without affecting their locomotion). Future work could explore methods for learning these disentangled representations from large-scale motion capture datasets using structured VAEs or other generative architectures that explicitly factorize motion into a semantic grammar of actions, styles, and body parts.

Ultimately, progress in these areas will be pivotal in making MoCap truly accessible to everyone. This, in turn, will unlock new opportunities that have yet to emerge due to the significant access constraints of such technologies.

References

[1] Matthew Loper, Naureen Mahmood, and Michael J Black. Mosh: motion and shape capture from sparse markers. *ACM Trans. Graph.*, 33(6):220–1, 2014.

[2] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proc. IEEE/CVF international conference on computer vision (CVPR)*, pages 5442–5451, 2019.

[3] L. McInnes, J. Healy, and J. Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[4] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019.

[5] Yinghao Huang, Federica Bogo, Christoph Lassner, Angjoo Kanazawa, Peter V Gehler, Javier Romero, Ijaz Akhter, and Michael J Black. Towards accurate marker-less human shape and pose estimation over time. In *2017 international conference on 3D vision (3DV)*, pages 421–430, 2017.

[6] Anurag Arnab, Carl Doersch, and Andrew Zisserman. Exploiting temporal context for 3d human pose estimation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3395–3404, 2019.

[7] Georgios Albanis, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis, and Kostas Kolomvatsos. Noise-in, bias-out: Balanced and real-time mocap solving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4237–4247, 2023.

[8] Buzhen Huang, Yuan Shu, Tianshu Zhang, and Yangang Wang. Dynamic multi-person mesh recovery from uncalibrated multi-view cameras. In *2021 International Conference on 3D Vision (3DV)*, pages 710–720. IEEE, 2021.

[9] Vickie Ye, Georgios Pavlakos, Jitendra Malik, and Angjoo Kanazawa. Decoupling human and camera motion from videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21222–21232, 2023.

[10] Georgios Albanis, Nikolaos Zioulis, and Kostas Kolomvatsos. Bundlemocap: Efficient, robust and smooth motion capture from sparse multiview videos. In *Proceedings of the 20th ACM SIGGRAPH European Conference on Visual Media Production*, pages 1–9, 2023.

[11] Jiawei Ren, Mingyuan Zhang, Cunjun Yu, and Ziwei Liu. Balanced mse for imbalanced visual regression. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7926–7935, 2022.

[12] Anargyros Chatzitofis, Dimitrios Zarpalas, Petros Daras, and Stefanos Kollias. Democap: low-cost marker-based motion capture. *International Journal of Computer Vision (IJCV)*, 129(12):3338–3366, 2021.

[13] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7718–7727, 2019.

[14] 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.

[15] 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.

[16] Hanz Cuevas-Velasquez, Anastasios Yiannakidis, Soyong Shin, Giorgio Becherini, Markus Höschle, Joachim Tesch, Taylor Obersat, Tsvetelina Alexiadis, and Michael J Black. Mamma: Markerless & automatic multi-person motion action capture. *arXiv preprint arXiv:2506.13040*, 2025.

[17] Peter J. Huber and Elvezio M. Ronchetti. *Robust Statistics*. Wiley Series in Probability and Statistics. John Wiley & Sons, 2009. ISBN 978-0-470-12990-6. doi: 10.1002/9780470434697.

[18] Donald Geman and Stuart Geman. Bayesian image analysis. In *Disordered systems and biological organization*, pages 301–319. Springer, 1986.

[19] Jonathan T Barron. A general and adaptive robust loss function. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4331–4339, 2019.

[20] Georgios Albanis, Anargyros Chatzitofis, Spyridon Thermos, Nikolaos Zioulis, and Kostas Kolomvatsos. Towards scalable and real-time markerless motion capture. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 724–725. IEEE, 2022.

[21] Georgios Albanis, Nikolaos Zioulis, and Kostas Kolomvatsos. Bundlemocap++: Efficient, robust and smooth motion capture from sparse multiview videos. *Computer Vision and Image Understanding*, 249:104190, 2024. ISSN 1077-3142. doi: https://doi.org/10.1016/j.cviu.2024.104190. URL `https://www.sciencedirect.com/science/article/pii/S1077314224002716`.

[22] Georgios Albanis, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis, and Kostas Kolomvatsos. From bias to balance: Leverage representation learning for bias-free mocap solving. *Computer Vision and Image Understanding*, 251:104241, 2025.

[23] Georgios Albanis, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis, and Kostas Kolomvatsos. Robust and efficient ai motion capture. In *DOCTORAL CONSORTIUM*, page 57, 2025.

[24] Georgios Albanis, Nikolaos Zioulis, Spyridon Thermos, Anargyros Chatzitofis, and Kostas Kolomvatsos. Mocatalyst: Accelerating and automating mocap. In *IEEE/CVF International Conference on Computer Vision Demos*, pages 1–2, 2023.

[25] Georgios Albanis, Nikolaos Zioulis, Anargyros Chatzitofis, Spyridon Thermos, Vladimiros Sterzentsenko, and Kostas Kolomvatsos. Lightmocap: Light-weight, real-time & scalable markerless motion capture. In *20th ACM SIGGRAPH European Conference on Visual Media Production Demos*, pages 1–2, 2023.

[26] Marilyn Keller, Keenon Werling, Soyong Shin, Scott Delp, Sergi Pujades, C Karen Liu, and Michael J Black. From skin to skeleton: Towards biomechanically accurate 3d digital humans. *ACM Transactions on Graphics (TOG)*, 42 (6):1–12, 2023.

[27] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM Transactions on Graphics (TOG)*, 34(6):1–16, 2015.

[28] David A Winter. *Biomechanics and motor control of human gait: normal, elderly and pathological*. University of Waterloo Press, 1991.

[29] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM Siggraph 2005 Papers*, pages 408–416. ASM, 2005.

[30] Emico Okuno and Luciano Fratin. *Biomechanics of the human body*, volume 1461485754. Springer, 2014.

[31] David Marr and H Keith Nishihara. Representation and recognition of the spatial organization of three-dimensional shapes. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 200(1140):269–294, 1978.

[32] J. P. Lewis, Matt Cordner, and Nickson Fong. Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 165–172, 2000.

[33] Brett Allen, Brian Curless, and Zoran Popović. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 147–156, 2006.

[34] Ilya Baran and Jovan Popović. Automatic rigging and animation of 3d characters. *ACM Transactions on graphics (TOG)*, 26(3):72–es, 2007.

[35] Zhan Xu, Yang Zhou, Evangelos Kalogerakis, Chris Landreth, and Karan Singh. Rignet: Neural rigging for articulated characters. *arXiv preprint arXiv:2005.00559*, 2020.

[36] Tobias Heimann and Hans-Peter Meinzer. Statistical shape models for 3d medical image segmentation: A review. *Medical Image Analysis*, 13(4):543–563, 2009.

[37] Timothy F Cootes, Christopher J Taylor, David H Cooper, and Jim Graham. Active shape models-their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[38] Daniel Rueckert, Alejandro F Frangi, and Julia A Schnabel. Automatic construction of 3-d statistical deformation models of the brain using nonrigid registration. *IEEE Transactions on Medical Imaging*, 22(8):1014–1025, 2003.

[39] 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.

[40] Ahmed A. A. Osman, Timo Bolkart, and Michael J. Black. Star: Sparse trained articulated human body regressor. In *European Conference on Computer Vision*, pages 598–613. Springer, 2020.

[41] Chun-Hao P. Huang, Hongwei Yi Li, and Michael J. Black. Supr: A sparse unified part-based human body model. In *European Conference on Computer Vision*, pages 608–626. Springer, 2022.

[42] Aaron Ferguson, Ahmed AA Osman, Berta Bescos, Carsten Stoll, Chris Twigg, Christoph Lassner, David Otte, Eric Vignola, Fabian Prada, Federica Bogo, et al. Mhr: Momentum human rig. *arXiv preprint arXiv:2511.15586*, 2025.

[43] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '87, pages 163–169. ACM, 1987. doi: 10.1145/37401.37422.

[44] Yiyi Liao, Simon Donné, and Andreas Geiger. Deep marching cubes: Learning explicit surface representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[45] Edoardo Remelli, Artem Lukoianov, Stephan R. Richter, Benoît Guillard, Timur Bagautdinov, Pierre Baque, and Pascal Fua. Meshsdf: Differentiable iso-surface extraction. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS)*, Vancouver, Canada, 2020.

[46] Stanley Osher and Nikos Paragios, editors. *Geometric Level Set Methods in Imaging, Vision, and Graphics*. Springer, New York, NY, USA, 2003. ISBN 978-0-387-95488-2.

[47] Lars M. Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision*

and Pattern Recognition (CVPR), pages 4455–4465, 2019. doi: 10.1109/CVPR.2019.00459.

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

[49] Shaohui Liu, Yinda Zhang, Songyou Peng, Boxin Shi, Marc Pollefeys, and Zhaopeng Cui. Dist: Rendering deep implicit signed distance function with differentiable sphere tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[50] Peng-Shuai Wang, Yang Liu, Yu-Qi Yang, and Xin Tong. Spline positional encoding for learning 3d implicit signed distance fields. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI-21)*, pages 1091–1097, 2021. doi: 10.24963/ijcai.2021/151.

[51] Songyou Peng, Chiyu Max Jiang, Yiyi Liao, Michael Niemeyer, Marc Pollefeys, and Andreas Geiger. Shape as points: A differentiable poisson solver. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

[52] Rohan Chabra, Jan E. Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *arXiv preprint arXiv:2003.10983*, 2020.

[53] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[54] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pages 3747–3757, Virtual, Online, 2020. PMLR. doi: 10.5555/3524938.3525293.

[55] Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20299–20309, 2024.

[56] Xian Liu, Xiaohang Zhan, Jiaxiang Tang, Ying Shan, Gang Zeng, Dahua Lin, Xihui Liu, and Ziwei Liu. Humangaussian: Text-driven 3d human generation with gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6646–6657, 2024.

[57] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20310–20320, 2024.

[58] Yufan Wu, Xuanhong Chen, Wen Li, Shunran Jia, Hualiang Wei, Kairui Feng, Jialiang Chen, Yuhan Li, Ang He, Weimin Zhang, et al. Sings: animatable single-image human gaussian splats with kinematic priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5571–5580, 2025.

[59] Junjin Xiao, Qing Zhang, Yonewei Nie, Lei Zhu, and Wei-Shi Zheng. Rogsplat: Learning robust generalizable human gaussian splatting from sparse multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5980–5990, 2025.

[60] Hong Xie, Xiaoyan Zhang, and Yingying Zhu. Nsghg: Neural surface guided generalizable human gaussian splatting for sparse view synthesis. *Neurocomputing*, 653:131207, 2025.

[61] Yingzhi Tang, Qijian Zhang, and Junhui Hou. Hugdiffusion: Generalizable single-image human rendering via 3d gaussian diffusion. *IEEE Transactions on Visualization and Computer Graphics*, 2025.

[62] Antonio Torralba, Phillip Isola, and William T Freeman. *Foundations of computer vision*. MIT Press, 2024.

[63] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Smplify: Automatic parameterization of 3d human body shape and pose from images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4800–4810, 2016.

[64] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J Black, and Peter V Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6050–6059, 2017.

[65] Nilesh Kulkarni, Abhinav Gupta, David F Fouhey, and Shubham Tulsiani. Articulation-aware canonical surface mapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 452–461, 2020.

[66] Siwei Zhang, Yan Zhang, Federica Bogo, Marc Pollefeys, and Siyu Tang. Learning motion priors for 4d human body capture in 3d scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11343–11353, 2021.

[67] Igor Santesteban, Miguel A Otaduy, and Dan Casas. Snug: Self-supervised neural dynamic garments. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8140–8150, 2022.

[68] Yunhai Xiao, Zengxin Wei, and Zhiguo Wang. A limited memory bfgs-type method for large-scale unconstrained optimization. *Computers & Mathematics with Applications*, 56(4):1001–1009, 2008.

[69] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017.

[70] Rawal Khirodkar, Timur Bagautdinov, Julieta Martinez, Su Zhaoen, Austin James, Peter Selednik, Stuart Anderson, and Shunsuke Saito. Sapiens: Foundation for human vision models. In *European Conference on Computer Vision*, pages 206–228. Springer, 2024.

[71] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[72] Apratim Bhattacharyya, Shweta Mahajan, Mario Fritz, Bernt Schiele, and Stefan Roth. Normalizing flows with multi-scale autoregressive priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8415–8424, 2020.

[73] Zhengdi Yu, Simone Foti, Linguang Zhang, Amy Zhao, Cem Keskin, Stefanos Zafeiriou, and Tolga Birdal. Geometric neural distance fields for learning human motion priors. *arXiv preprint arXiv:2509.09667*, 2025.

[74] Guénolé Fiche, Simon Leglaive, Xavier Alameda-Pineda, Antonio Agudo, and Francesc Moreno-Noguer. Vq-hps: Human pose and shape estimation in a

vector-quantized latent space. In *European Conference on Computer Vision*, pages 471–490. Springer, 2024.

[75] Junzhe Lu, Jing Lin, Hongkun Dou, Ailing Zeng, Yue Deng, Xian Liu, Zhongang Cai, Lei Yang, Yulun Zhang, Haoqian Wang, et al. Dposer-x: Diffusion model as robust 3d whole-body human pose prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9988–9997, 2025.

[76] Zhongyu Jiang, Zhuoran Zhou, Lei Li, Wenhao Chai, Cheng-Yen Yang, and Jenq-Neng Hwang. Back to optimization: Diffusion-based zero-shot 3d human pose estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6142–6152, 2024.

[77] Haoxin Yang, Weihong Chen, Xuemiao Xu, Cheng Xu, Peng Xiao, Cuifeng Sun, Shaoyu Huang, and Shengfeng He. Starpose: 3d human pose estimation via spatial-temporal autoregressive diffusion. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[78] Weiquan Wang, Jun Xiao, Chunping Wang, Wei Liu, Zhao Wang, and Long Chen. Dipose: Discrete diffusion model for occluded 3d human pose estimation. *Advances in Neural Information Processing Systems*, 37:98717–98741, 2024.

[79] Seoyoung Lee and Joonseok Lee. Posediff: Pose-conditioned multimodal diffusion model for unbounded scene synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5007–5017, 2024.

[80] Jinglin Xu, Yijie Guo, and Yuxin Peng. Finepose: Fine-grained prompt-driven 3d human pose estimation via diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 561–570, 2024.

[81] Mihai Fieraru, Mihai Zanfir, Silviu Cristian Pirlea, Vlad Olaru, and Cristian Sminchisescu. Aifit: Automatic 3d human-interpretable feedback models for fitness training. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9919–9928, 2021.

[82] Joachim Tesch, Giorgio Becherini, Prerana Achar, Anastasios Yiannakidis, Muhammed Kocabas, Priyanka Patel, and Michael J Black. Bedlam2. 0: Synthetic humans and cameras in motion. *arXiv preprint arXiv:2511.14394*, 2025.

[83] Yu Rong, Ziwei Liu, and Chen Change Loy. Chasing the tail in monocular 3d human reconstruction with prototype memory. *IEEE Transactions on Image Processing (TIP)*, 31:2907–2919, 2022.

[84] Luís Torgo, Paula Branco, Rita P Ribeiro, and Bernhard Pfahringer. Resampling strategies for regression. *Expert Systems*, 32(3):465–476, 2015.

[85] Luís Torgo, Rita P Ribeiro, Bernhard Pfahringer, and Paula Branco. Smote for regression. In *Progress in Artificial Intelligence: 16th Portuguese Conference on Artificial Intelligence, EPIA 2013, Angra do Heroísmo, Azores, Portugal, September 9-12, 2013. Proceedings 16*, pages 378–389. Springer, 2013.

[86] Paula Branco, Rita P Ribeiro, and Luis Torgo. Ubl: an r package for utility-based learning. *arXiv preprint arXiv:1604.08079*, 2016.

[87] Paula Branco, Luís Torgo, and Rita P Ribeiro. SMOGN: A pre-processing approach for imbalanced regression. In *First international workshop on learning with imbalanced domains: Theory and applications*, pages 36–50. PMLR, 2017.

[88] Luis Torgo and Rita Ribeiro. Utility-based regression. In *PKDD*, volume 7, pages 597–604. Springer, 2007.

[89] Michael Steininger, Konstantin Kobs, Padraig Davidson, Anna Krause, and Andreas Hotho. Density-based weighting for imbalanced regression. *Machine Learning*, 110:2187–2211, 2021.

[90] Aníbal Silva, Rita P Ribeiro, and Nuno Moniz. Model optimization in imbalanced regression. In *Discovery Science: 25th International Conference, DS 2022, Montpellier, France, October 10–12, 2022, Proceedings*, pages 3–21. Springer, 2022.

[91] Yuzhe Yang, Kaiwen Zha, Yingcong Chen, Hao Wang, and Dina Katabi. Delving into deep imbalanced regression. In *Proc. International Conference on Machine Learning (ICML)*, pages 11842–11851. PMLR, 2021.

[92] Yu Gong, Greg Mori, and Frederick Tung. RankSim: Ranking similarity regularization for deep imbalanced regression. *arXiv preprint arXiv:2205.15236*, 2022.

[93] Leonid Sigal, Alexandru Balan, and Michael Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. *Advances in neural information processing systems*, 20, 2007.

[94] Peng Guan, Alexander Weiss, Alexandru O Balan, and Michael J Black. Estimating human shape and pose from a single image. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1381–1388. IEEE, 2009.

[95] 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.

[96] Zhongguo Li, Magnus Oskarsson, and Anders Heyden. 3d human pose and shape estimation through collaborative learning and multi-view model-fitting. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1888–1897, 2021.

[97] Zijian Dong, Jie Song, Xu Chen, Chen Guo, and Otmar Hilliges. Shape-aware multi-person pose estimation from multi-view images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11158–11168, 2021.

[98] Junbang Liang and Ming C Lin. Shape-aware human pose and shape reconstruction using multi-view images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4352–4362, 2019.

[99] Davis Rempe, Jun Wang, Angjoo Kanazawa, Vladlen Koltun, and Helge Rhodin. Humor: 3d human motion model for robust pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[100] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4501–4510, 2019.

[101] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1954–1963, 2021.

[102] Ziwei Liao, Jialiang Zhu, Chunyu Wang, Han Hu, and Steven L Waslander. Multiple view geometry transformers for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 708–717, 2024.

[103] Yu-Pei Song, Xiao Wu, Zhaoquan Yuan, Jian-Jun Qiao, and Qiang Peng. Posturehmr: Posture transformation for 3d human mesh recovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9732–9741, 2024.

[104] Qingping Sun, Yanjun Wang, Ailing Zeng, Wanqi Yin, Chen Wei, Wenjia Wang, Haiyi Mei, Chi-Sing Leung, Ziwei Liu, Lei Yang, et al. Aios: All-in-one-stage expressive human pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1834–1843, 2024.

[105] Wanqi Yin, Zhongang Cai, Ruisi Wang, Ailing Zeng, Chen Wei, Qingping Sun, Haiyi Mei, Yanjun Wang, Hui En Pang, Mingyuan Zhang, et al. Smplest-x: Ultimate scaling for expressive human pose and shape estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[106] Soyong Shin, Juyong Kim, Eni Halilaj, and Michael J Black. Wham: Reconstructing world-grounded humans with accurate 3d motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2070–2080, 2024.

[107] Yufu Wang, Ziyun Wang, Lingjie Liu, and Kostas Daniilidis. Tram: Global trajectory and motion of 3d humans from in-the-wild videos. In *European Conference on Computer Vision*, pages 467–487. Springer, 2024.

[108] Yizhou Zhao, Tuanfeng Yang Wang, Bhiksha Raj, Min Xu, Jimei Yang, and Chun-Hao Paul Huang. Synergistic global-space camera and human reconstruction from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1216–1226, 2024.

[109] 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 Conference Papers*, pages 1–11, 2024.

[110] Yuhong Zhang, Guanlin Wu, Ling-Hao Chen, Zhuokai Zhao, Jing Lin, Xiaoke Jiang, Jiamin Wu, Zhuoheng Li, Hao Frank Yang, Haoqian Wang, et al. Humanmm: Global human motion recovery from multi-shot videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1973–1983, 2025.

[111] 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.

[112] Romain Brégier, Fabien Baradel, Thomas Lucas, Salma Galaaoui, Matthieu Armando, Philippe Weinzaepfel, and Grégory Rogez. Condimen: Conditional

multi-person mesh recovery. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3880–3890, 2025.

[113] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.

[114] Yaguang Yang. Globally convergent optimization algorithms on riemannian manifolds: Uniform framework for unconstrained and constrained optimization. *Journal of Optimization Theory and Applications*, 132:245–265, 2007.

[115] Navin Goyal and Abhishek Shetty. Sampling and optimization on convex sets in riemannian manifolds of non-negative curvature. In *Conference on Learning Theory*, pages 1519–1561. PMLR, 2019.

[116] Wolfgang Ring and Benedikt Wirth. Optimization methods on riemannian manifolds and their application to shape space. *SIAM Journal on Optimization*, 22(2):596–627, 2012.

[117] P-A Absil, Christopher G Baker, and Kyle A Gallivan. Trust-region methods on riemannian manifolds. *Foundations of Computational Mathematics*, 7(3): 303–330, 2007.

[118] Chunhong Qi, Kyle A Gallivan, and P-A Absil. Riemannian bfgs algorithm with applications. In *Recent Advances in Optimization and its Applications in Engineering: The 14th Belgian-French-German Conference on Optimization*, pages 183–192. Springer, 2010.

[119] Wen Huang, Kyle A Gallivan, and P-A Absil. A broyden class of quasi-newton methods for riemannian optimization. *SIAM Journal on Optimization*, 25(3): 1660–1685, 2015.

[120] Christian Keilstrup Ingwersen, Christian Møller Mikkelstrup, Janus Nørtoft Jensen, Morten Rieger Hannemose, and Anders Bjorholm Dahl. Sportspose-a dynamic 3d sports pose dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5218–5227, 2023.

[121] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5614–5623, 2019.

[122] Runze Li, Srikrishna Karanam, Ren Li, Terrence Chen, Bir Bhanu, and Ziyan Wu. Learning local recurrent models for human mesh recovery. In *2021 International Conference on 3D Vision (3DV)*, pages 555–564. IEEE, 2021.

[123] Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. Sfv: Reinforcement learning of physical skills from videos. *ACM Transactions On Graphics (TOG)*, 37(6):1–14, 2018.

[124] Xiuming Zhang, Tali Dekel, Tianfan Xue, Andrew Owens, Qiurui He, Jiajun Wu, Stefanie Mueller, and William T Freeman. Mosculp: Interactive visualization of shape and time. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, pages 275–285, 2018.

[125] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. Monocular 3d pose and shape estimation of multiple people in natural scenes-the importance of multiple scene constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2148–2157, 2018.

[126] Yinghao Huang, Omid Taheri, Michael J Black, and Dimitrios Tzionas. Intercap: Joint markerless 3d tracking of humans and objects in interaction. In *DAGM German Conference on Pattern Recognition*, pages 281–299. Springer, 2022.

[127] Nitin Saini, Chun-Hao P Huang, Michael J Black, and Aamir Ahmad. Smartmocap: Joint estimation of human and camera motion using uncalibrated rgb cameras. *IEEE Robotics and Automation Letters*, 2023.

[128] Pengle Jin and Xinguo Liu. Robust human motion estimation using bidirectional motion prior model and spatiotemporal progressive motion optimization. *Computers & Graphics*, 2023.

[129] Ijaz Akhter, Tomas Simon, Sohaib Khan, and Yaser Sheikh. Pose-dependent joint angle limits for 3d human pose reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4800–4810, 2012.

[130] Guang Yang, Xinyu Huang, Sunil Lim, Yichao Wang, He Fang, Khoa Luu, and Thien Huynh Nguyen. Pose guided human image generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[131] Yan Zhang, Zhen Yang, Ruojin Tang, Ping Wei, Xin Sun, Lizhuang Ma, and Xiaokang Tong. Learning 3d human shape and pose from dense body parts. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[132] Jiangning Wang, Jianfeng Zhang, Fei Gao, Feng Liu, and Hongwei Liu. Mvae: Multimodal variational autoencoder for human motion sequence generation. In

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

[133] Mathis Petrovich, Michael J Black, and Gül Varol. Actor: Learning motion priors for 3d human animation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[134] 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 (CVPR)*, 2020.

[135] Hai Ci, Mingdong Wu, Wentao Zhu, Xiaoxuan Ma, Hao Dong, Fangwei Zhong, and Yizhou Wang. Gfpose: Learning 3d human pose prior with gradient fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4800–4810, 2023.

[136] Dieederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations (ICLR)*, 2015.

[137] Emile Mathieu, Charline Le Lan, Chris J Maddison, Ryota Tomioka, and Yee Whye Teh. Continuous hierarchical representations with poincaré variational auto-encoders. *Advances in neural information processing systems*, 32, 2019.

[138] Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M Tomczak. Hyperspherical variational auto-encoders. *arXiv preprint arXiv:1804.00891*, 2018.

[139] Jiacheng Xu and Greg Durrett. Spherical latent spaces for stable variational autoencoders. *arXiv preprint arXiv:1808.10805*, 2018.

[140] Luca Falorsi, Pim De Haan, Tim R Davidson, Nicola De Cao, Maurice Weiler, Patrick Forré, and Taco S Cohen. Explorations in homeomorphic variational auto-encoding. *arXiv preprint arXiv:1807.04689*, 2018.

[141] Hang Shao, Abhishek Kumar, and P Thomas Fletcher. The riemannian geometry of deep generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 315–323, 2018.

[142] Deli Zhao, Jiapeng Zhu, and Bo Zhang. Latent variables on spheres for autoencoders in high dimensions. arxiv. *Learning*, 2019.

[143] Jack B Kuipers. *Quaternions and rotation sequences: a primer with applications to orbits, aerospace, and virtual reality*. Princeton university press, 1999.

[144] Tatiana Shingel. Interpolation in special orthogonal groups. *IMA journal of numerical analysis*, 29(3):731–745, 2009.

[145] Ki Wai Fong and Shingyu Leung. Spherical essentially non-oscillatory (seno) interpolation. *Journal of Scientific Computing*, 94(1):28, 2023.

[146] Raymond Yeh, Ziwei Liu, Dan B Goldman, and Aseem Agarwala. Semantic facial expression editing using autoencoded flow. *arXiv preprint arXiv:1611.09961*, 2016.

[147] Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*, 2015.

[148] Andrey Davydov, Anastasia Remizova, Victor Constantin, Sina Honari, Mathieu Salzmann, and Pascal Fua. Adversarial parametric pose prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10997–11005, 2022.

[149] David Berthelot, Colin Raffel, Aurko Roy, and Ian Goodfellow. Understanding and improving interpolation in autoencoders via an adversarial regularizer. *arXiv preprint arXiv:1807.07543*, 2018.

[150] Saeed Ghorbani, Ali Etemad, and Nikolaus F Troje. Auto-labelling of markers in optical motion capture by permutation learning. In *Advances in Computer Graphics: 36th Computer Graphics International Conference, CGI 2019, Calgary, AB, Canada, June 17–20, 2019, Proceedings 36*, pages 167–178. Springer, 2019.

[151] Nima Ghorbani and Michael J Black. Soma: Solving optical marker-based mocap automatically. In *Proc. IEEE/CVF International Conference on Computer Vision (CVPR)*, pages 11117–11126, 2021.

[152] Shangchen Han, Beibei Liu, Robert Wang, Yuting Ye, Christopher D Twigg, and Kenrick Kin. Online optical marker-based hand tracking with deep labels. *ACM Transactions on Graphics (TOG)*, 37(4):1–10, 2018.

[153] Kang Chen, Yupan Wang, Song-Hai Zhang, Sen-Zhe Xu, Weidong Zhang, and Shi-Min Hu. Mocap-solver: A neural solver for optical motion capture data. *ACM Transactions on Graphics (TOG)*, 40(4):1–11, 2021.

[154] Daniel Holden. Robust solving of optical motion capture data by denoising. *ACM Transactions on Graphics (TOG)*, 37(4):1–12, 2018.

[155] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *Proc. International Conference on Learning Representations, ICLR*, pages 1–15, 2014.

[156] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proc. International Conference on Machine Learning (ICML)*, pages 1530–1538. PMLR, 2015.

[157] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[158] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[159] Taylor Denouden, Rick Salay, Krzysztof Czarnecki, Vahdat Abdelzad, Buu Phan, and Sachin Vernekar. Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance. *arXiv preprint arXiv:1812.02765*, 2018.

[160] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[161] Ken Shoemake. Animating rotation with quaternion curves. In *Proc. Conference on Computer Graphics and Interactive Techniques*, page 245–254, 1985.

[162] Ali Jahanian, Lucy Chai, and Phillip Isola. On the "steerability" of generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2020.

[163] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4401–4410, 2019.

[164] Tom White. Sampling generative networks: Notes on a few effective techniques. *arXiv:1609.04468*, 2016.

[165] Aiden Nibali, Zhen He, Stuart Morgan, and Luke Prendergast. 3d human pose estimation with 2d marginal heatmaps. In *Proc. IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1477–1485. IEEE, 2019.

[166] Hang Ye, Wentao Zhu, Chunyu Wang, Rujie Wu, and Yizhou Wang. Faster voxelpose: Real-time 3d human pose estimation by orthographic projection. In *Proc. European Conference on Computer Vision (ECCV)*, pages 142–159. Springer, 2022.

[167] Diogo C Luvizon, Hedi Tabia, and David Picard. Human pose regression by combining indirect part detection and contextual information. *Computers & Graphics*, 85:15–22, 2019.

[168] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proc. European conference on computer vision (ECCV)*, pages 529–545, 2018.

[169] Aiden Nibali, Zhen He, Stuart Morgan, and Luke Prendergast. Numerical coordinate regression with convolutional neural networks. *arXiv preprint arXiv:1801.07372*, 2018.

[170] Christopher Tensmeyer and Tony Martinez. Robust keypoint detection. In *Proc. International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 5, pages 1–7, 2019. doi: 10.1109/ICDARW.2019.40072.

[171] ML Menéndez, JA Pardo, L Pardo, and MC Pardo. The jensen-shannon divergence. *Journal of the Franklin Institute*, 334(2):307–318, 1997.

[172] Paul W Holland and Roy E Welsch. Robust regression using iteratively reweighted least-squares. *Communications in Statistics-theory and Methods*, 6(9):813–827, 1977.

[173] John E Dennis Jr and Roy E Welsch. Techniques for nonlinear least squares and robust regression. *Communications in Statistics-simulation and Computation*, 7(4):345–359, 1978.

[174] Andrey Davydov, Anastasia Remizova, Victor Constantin, Sina Honari, Mathieu Salzmann, and Pascal Fua. Adversarial parametric pose prior. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10987–10995, 2022.

[175] Garvita Tiwari, Dimitrije Antić, Jan Eric Lenssen, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. Pose-ndf: Modeling human pose manifolds with neural distance fields. In *Proc. European Conference on Computer Vision (ECCV)*, pages 572–589. Springer, 2022.

[176] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.

[177] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proc. IEEE International Conference on Image Processing (ICIP)*, pages 168–172, 1994.

[178] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proc. International Conference on Learning Representations, (ICLR)*, pages 1–15, 2019.

[179] Ijaz Akhter and Michael J Black. Pose-conditioned joint angle limits for 3D human pose reconstruction. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1446–1455, 2015.

[180] Leonid Sigal, Alexandru O Balan, and Michael J Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision*, 87 (1-2):4, 2010.

[181] Advanced Computing Center for the Arts and Design. ACCAD MoCap Dataset, 2017. URL `https://accad.osu.edu/research/motion-lab/mocap-system-and-data`.

[182] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 8320–8329, 2018.

[183] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Dynamic FAUST: Registering human bodies in motion. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6233–6242, 2017.

[184] Tao Yu, Zerong Zheng, Kaiwen Guo, Pengpeng Liu, Qionghai Dai, and Yebin Liu. Function4d: Real-time human volumetric capture from very sparse consumer rgbd sensors. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5746–5756, 2021.

[185] Wei Cheng, Su Xu, Jingtan Piao, Chen Qian, Wayne Wu, Kwan-Yee Lin, and Hongsheng Li. Generalizable neural performer: Learning robust radiance fields for human novel view synthesis. *arXiv preprint arXiv:2204.11798*, 2022.

[186] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2Motion: Conditioned generation of 3D human motions. In *Proc. ACM International Conference on Multimedia (MM)*, page 2021–2029, 2020.

[187] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7482–7491, 2018.

[188] Lukas Neumann, Tri Pham, Theodora Sarlos, and Mykhaylo Vladymyrov. A student-t mixture likelihood for robust optimization. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *PMLR*, pages 8066–8075, 2021.

[189] Tom Wehrbein, Marco Rudolph, Bodo Rosenhahn, and Bastian Wandt. Utilizing uncertainty in 2d pose detectors for probabilistic 3d human mesh recovery. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5852–5862. IEEE, 2025.

[190] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 13(4):376–380, 2002.

[191] Rainer Kümmerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g 2 o: A general framework for graph optimization. In *2011 IEEE international conference on robotics and automation*, pages 3607–3613. IEEE, 2011.

[192] Yating Tian, Hongwen Zhang, Yebin Liu, and Limin Wang. Recovering 3d human mesh from monocular images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[193] Wen-Li Wei, Jen-Chun Lin, Tyng-Luh Liu, and Hong-Yuan Mark Liao. Capturing humans in motion: Temporal-attentive 3d human pose and shape estimation from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13211–13220, 2022.

[194] Ailing Zeng, Lei Yang, Xuan Ju, Jiefeng Li, Jianyi Wang, and Qiang Xu. Smoothnet: A plug-and-play network for refining human poses in videos. In *European Conference on Computer Vision*, pages 625–642. Springer, 2022.

[195] Zhenhua Tang, Zhaofan Qiu, Yanbin Hao, Richang Hong, and Ting Yao. 3d human pose estimation with spatio-temporal criss-cross attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4790–4799, 2023.

[196] Xiaolong Shen, Zongxin Yang, Xiaohan Wang, Jianxin Ma, Chang Zhou, and Yi Yang. Global-to-local modeling for video-based 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8887–8896, 2023.

[197] Matteo Ruggero Ronchi and Pietro Perona. Benchmarking and error diagnosis in multi-instance pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 369–378, 2017.

[198] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[199] moai: PyTorch Model Development Kit. *moai*: Accelerating modern data-driven workflows. `https://github.com/ai-in-motion/moai`, 2021.

[200] N Miolane, J Mathe, C Donnat, and M Jorda. Pennec, x. geomstats: a python package for riemannian geometry in machine learning. *arXiv preprint arXiv:1805.08308*, 2018.

[201] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014.

[202] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017. doi: 10.1109/3dv.2017.00064. URL `http://gvv.mpi-inf.mpg.de/3dhp_dataset`.

[203] Matthew Shere, Hansung Kim, and Adrian Hilton. Temporally consistent 3d human pose estimation using dual 360deg cameras. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 81–90, 2021.

[204] Aymen Mir, Xavier Puig, Angjoo Kanazawa, and Gerard Pons-Moll. Generating continual human motion in diverse 3d scenes. *arXiv preprint arXiv:2304.02061*, 2023.