SynHMR: Synergistic Joint-Mesh Modeling for LiDAR-based Human Mesh Reconstruction

Rui Shi[1†], Xiaoqi An[1†], Lin Zhao[1†*], Di Wang[2], Chen Gong[3], and Le Zhang[4]

[1] School of Computer Science and Engineering, Nanjing University of Science and Technology, China
[2] School of Computer Science and Technology, Xidian University, China
[3] School of Automation and Intelligent Sensing, Shanghai Jiao Tong University, China
[4] School of Information and Communication Engineering, University of Electronic Science and Technology of China, China
{124106010810, xiaoqi.an, linzhao}@njust.edu.cn, wangdi@xidian.edu.cn, chen.gong@sjtu.edu.cn, lezhang@uestc.edu.cn

Abstract. LiDAR-based Human Mesh Reconstruction (HMR) is crucial for understanding human activities in real-world environments. However, the inherent sparsity of LiDAR point clouds often causes severe loss of anatomical details, limiting reconstruction accuracy. Although recent methods have made promising progress, most still follow a decoupled pipeline that first estimates skeletal joints and then reconstructs the mesh, leading to error propagation under sparse observations. To address this issue and fully exploit both the topological guidance of the skeleton on the mesh and the spatial constraints of the surface mesh on the skeleton, we propose SynHMR, a Synergistic Joint-Mesh Modeling framework for robust LiDAR-based HMR. Specifically, we first initialize both the skeletal joints and the mesh vertices as queries equipped with positional information for co-optimization. We then design a Geometry-Aware Transformer that leverages point cloud features to apply the attention mechanism between joint and vertex queries, enabling their iterative refinement through mutual information exchange. Furthermore, we introduce a Noise-Augmented Learning strategy that injects perturbed joint queries during training. This reduces the impact of noise while encouraging bidirectional association between joints and mesh. Extensive experiments on LiDARHuman26M, SLOPER4D, and Human-M3 demonstrate that SynHMR achieves state-of-the-art performance among LiDAR-based HMR approaches. The project page is available at https://github.com/ShiRui1208/SynHMR.

Keywords: 3D human mesh reconstruction · sparse point clouds · deep learning

---

†: Equal contribution, *: Corresponding author.

1 Introduction

3D Human Mesh Reconstruction (HMR) in complex 3D environments [27, 36] has emerged as a fundamental task for understanding human behavior. This task aims to predict a 3D human model from observed data, either as a non-parametric mesh [19] or as a parametric description [13, 15] derived from statistical human body models [2, 28, 30]. Compared to 3D Human Pose Estimation (HPE), HMR provides additional information regarding human shape and surface details, extending its utility to a broader range of applications such as AR/VR [37], robotics [32], and autonomous driving [23].

Traditional human pose estimation (HPE) and human mesh reconstruction (HMR) methods that rely on RGB [5,14,22,24,25] or RGB-D [3,4,11,43] sensors suffer from several limitations, such as poor robustness under complex illumination, risks of privacy leakage [12], and restricted detection ranges in indoor environments. In contrast, LiDAR provides a robust, long-range, and privacy-preserving [6, 34, 39] sensing modality well-suited for human-centric perception in real-world settings.

Although significant progress has been made [9,13,21,22,26,38], most LiDAR-based HMR methods, as illustrated in Fig. 1(a), adopt a decoupled approach that first estimates the skeletal pose and then reconstructs the mesh. This stepwise modeling is heavily challenged by the inherent sparsity of LiDAR point clouds. Estimating joints first fails to exploit local surface and shape cues to correct early joint errors. Consequently, the subsequent mesh reconstruction stage not only propagates but often amplifies these initial errors. While some approaches [9,38] adopt coarse-to-fine reconstruction frameworks, they still model pose and mesh separately, overlooking the spatial constraints that the surface mesh can impose on the skeletal joints.

To overcome the above limitations, we reconsider the relationship between skeletal joints and mesh vertices. Previous studies [9, 13, 21, 26] have shown that skeletal joints can provide strong topological guidance for human mesh reconstruction. Conversely, compared with sparse joint representations, the mesh surface is more directly constrained by local point-cloud observations and can therefore provide valuable evidence for correcting joint locations. Motivated by this synergy, as illustrated in Fig. 1(b), we represent the human body using both skeletal joints and a set of sampled mesh vertices, and jointly optimize them within a unified framework. This design breaks the conventional one-way skeleton-to-mesh dependency, enabling the model to exploit the synergistic relationship between the skeleton and the mesh.

Based on this synergistic joint-mesh representation, we propose SynHMR, a single-frame LiDAR-based HMR framework. Unlike decoupled pipelines that treat joint estimation merely as a prerequisite for mesh reconstruction, our framework formulates both the skeletal joints and the sampled mesh vertices as queries equipped with explicit positional information. These queries are then jointly optimized to break the conventional one-way causal chain. We design a Geometry-Aware Transformer that allows these queries to continuously exchange information and mutually refine within a shared representation space. Specifi-

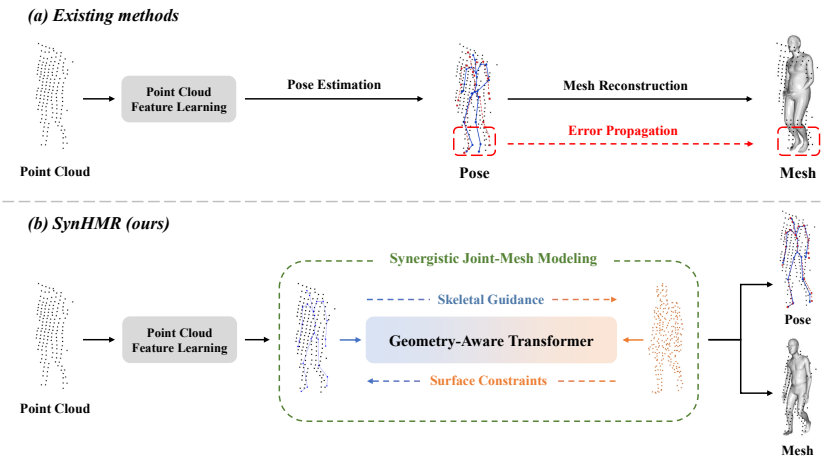

Fig. 1: Comparison between existing methods and SynHMR (ours). (a) Existing methods follow a decoupled pipeline, where pose is first estimated from sparse point clouds and mesh reconstruction is then conditioned on the estimated pose, causing errors to propagate under sparse, noisy, and incomplete observations. **(b)** SynHMR performs synergistic joint-mesh modeling via a Geometry-Aware Transformer, which jointly refines joints and vertices. Surface constraints improve joint estimation, while skeletal guidance stabilizes mesh reconstruction for robust human mesh recovery.

cally, through attention mechanisms, the mesh surface geometry provides shape-aware spatial constraints for correcting joint positions, while the skeletal structure guides mesh deformation to ensure biomechanical plausibility. Both types of queries adaptively retrieve local geometric cues from multi-level point features, effectively anchoring their mutual refinement in real 3D observations. This is particularly important in challenging regions such as limbs, where point clouds are often severely incomplete. Furthermore, we introduce a Noise-Augmented Learning strategy to strengthen the joint-mesh dependency during training. By perturbing joint queries with Gaussian noise, the network is encouraged to rely on joint-mesh synergy and local surface observations to correct erroneous poses, thereby improving robustness under sparse conditions. We conduct extensive experiments on LiDARHuman26M [21], SLOPER4D [7], and Human-M3 [8]. The results show that our method outperforms existing methods and achieves state-of-the-art performance on all three benchmarks, validating the effectiveness of synergistic joint-mesh modeling.

In summary, the contributions of this paper are as follows:

- We propose a synergistic joint-mesh modeling framework that enables bidirectional interaction between skeletal joints and mesh vertices, leveraging their synergy to handle sparse LiDAR observations.
- We introduce a noise-augmented learning strategy that explicitly encourages the model to learn the intrinsic spatial dependencies between the skeleton and the mesh, significantly improving the robustness of the model.

– The proposed method achieves state-of-the-art performance on three public datasets, namely LiDARHuman26M, SLOPER4D, and Human-M3.

2 Related Work

3D Human Pose Estimation from LiDAR Point Clouds. RGB-based 3D human mesh reconstruction methods [5, 15, 18, 22, 24, 25] are often limited by depth ambiguity and environmental variations. In contrast, LiDAR point clouds provide explicit 3D geometric observations, making them more suitable for robust human reconstruction in complex scenes. Recent studies explore various methods for 3D human pose estimation from LiDAR point clouds. Kovács et al. [20] combine a vision transformer with foreground-background segmentation for real-time pose estimation, while Zhang et al. [41] introduce a 3D scanning neighbor module to leverage environmental context for better pose learning. Multimodal fusion methods [6, 31, 42] further use 2D image annotations as weak supervision. Ye et al. [40] incorporate 3D detection and semantic segmentation features into pose estimation and employ a keypoint transformer for human keypoint prediction. An et al. [1] refine attention computation based on spatial density, improving performance on non-uniform point clouds.

However, these methods focus solely on estimating skeletal joints without reconstructing the human mesh, thereby lacking body shape details and missing crucial surface constraints to prevent biomechanically implausible poses.

3D Human Mesh Reconstruction from LiDAR Point Clouds. Research on 3D human mesh reconstruction from LiDAR point clouds remains limited, and existing methods mainly rely on dense point clouds [13, 26] or sequential point clouds [21, 38]. Jiang et al. [13] propose a skeleton-aware framework that maps point cloud features to skeletal joint features for SMPL parameter regression. Liu et al. [26] aggregate joint features from local point clouds via a voting mechanism to regress SMPL parameters. Li et al. [21] perform temporal encoding on sequential LiDAR point clouds and combine inverse kinematics with SMPL optimization for 3D human motion capture. Nevertheless, these methods do not effectively exploit local geometric details and thus perform poorly on sparse inputs. Wang et al. [38] and Fan et al. [9] further adopt coarse-to-fine strategies to progressively reconstruct human meshes from sequential or single-frame point clouds.

However, these methods mainly model how skeletal structure guides mesh reconstruction, while overlooking the spatial constraints of mesh geometry on skeletal estimation.

3 Method

To fully exploit the intrinsic relationship between the skeleton and the mesh in LiDAR-based HMR, we propose SynHMR, as illustrated in Fig. 2. SynHMR consists of three modules. First, a coarse joint-mesh query initializer extracts point cloud features and produces coarse joint and sampled vertex predictions, which

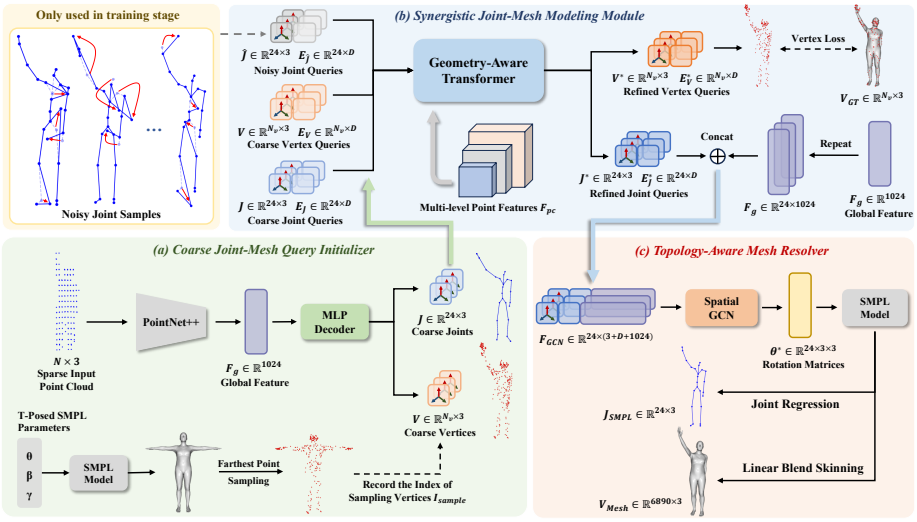

Fig. 2: Our framework mainly consists of three modules: (a) a coarse joint-mesh query initializer for extracting point cloud features and generating initial joint and vertex queries; (b) a synergistic joint-mesh modeling module that enables interaction between joint and vertex queries and iteratively refines them; and (c) a topology-aware mesh resolver for regressing the final SMPL body parameters.

are used to initialize the queries for subsequent joint-mesh refinement. Second, a synergistic joint-mesh modeling module jointly refines joints and vertices by exploring their geometric relationships in space. Finally, based on human anatomy and refined representations, we use a topology-aware mesh resolver to estimate SMPL parameters, obtaining joint positions and mesh simultaneously.

3.1 Coarse Joint-Mesh Query Initializer

Given a single-frame sparse LiDAR point cloud, the input is first normalized to satisfy the unified input requirement of the model. Following the preprocessing strategy in LiDARCap [21], the sparse point cloud is converted into a fixed-size point set $P \in \mathbb{R}^{N\times 3}$ with $N = 512$ via random sampling or repetition-based padding. We then employ a PointNet++ [33] encoder as the backbone to extract geometric representations from the sparse input. The encoder is composed of multiple hierarchical set abstraction (SA) layers, which progressively downsample the point cloud to capture multi-level point features. This process produces two types of features: (1) a set of multi-level point features F_{pc} extracted from intermediate layers, preserving fine-grained spatial details; and (2) a high-dimensional global feature $F_{\text{g}} \in \mathbb{R}^{1024}$, which aggregates global semantic information relevant to human pose.

Representative vertex sampling. Directly regressing all 6,890 vertices of the SMPL mesh is computationally expensive and inefficient, especially under

sparse observations. Therefore, we represent the body surface using a subset of N_v mesh vertices. Specifically, we perform farthest point sampling (FPS) on the standard SMPL mean template in the T-pose to ensure that the sampled vertices uniformly cover key body parts such as limbs and torso. The indices of these sampled vertices are recorded as I_{sample}.

Initial location estimation. We adopt a multilayer perceptron (MLP) decoder to initialize the spatial locations of joints and sampled vertices from the global feature F_{g}. As illustrated in Fig. 2(a), the decoder simultaneously predicts the 3D coordinates of 24 skeletal joints and the 3D coordinates of the N_v sampled vertices, which serve as the initial joint and vertex queries for the subsequent synergistic joint-mesh modeling module. The initial estimates are formulated as:

$$\begin{aligned} J &= \mathrm{MLP}_{\mathrm{joint}}(F_{\mathrm{g}}) \in \mathbb{R}^{24\times 3}, \\ V &= \mathrm{MLP}_{\mathrm{vertex}}(F_{\mathrm{g}}) \in \mathbb{R}^{N_v\times 3}, \end{aligned} \tag{1}$$

where J and V denote the initialized locations of skeletal joints and sampled vertices, respectively.

Noise-Augmented Learning Strategy. To further enhance the synergy between joints and mesh during training, inspired by Poseur [29], we introduce a noise-augmented learning strategy on the initialized joints. Specifically, Gaussian noise with standard deviation σ is added to the initialized joint coordinates to generate perturbed joint samples $\hat{J} = J + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$. These perturbed joints, together with the initialized joints and sampled vertices, are fed into the subsequent synergistic joint-mesh modeling module to construct the corresponding joint and vertex queries. By jointly refining initialized and perturbed joints with mesh vertices, the model can better exploit joint-mesh synergy and improve robustness to noise.

3.2 Synergistic Joint-Mesh Modeling Module

Most LiDAR-based 3D human mesh reconstruction methods decouple joint estimation and mesh reconstruction, which propagates errors under sparse and incomplete point clouds. To address this issue, we propose a synergistic joint-mesh modeling module. We first initialize a set of synergistic queries. Then, a Geometry-Aware Transformer is used to refine the joints and vertices under bidirectional geometric constraints. The Gaussian Fourier Positional Encoding (GFPE) module in each layer enables surface geometry to constrain skeletal joints while allowing skeletal topology to guide mesh deformation.

Synergistic Query Initialization. We initialize a set of learnable embeddings for the initialized joints, perturbed joints, and sampled vertices, denoted as $E_J \in \mathbb{R}^{24\times D}$, $E_{\hat{J}} \in \mathbb{R}^{24\times D}$, and $E_V \in \mathbb{R}^{N_v\times D}$, respectively, which serve as the content embeddings of the queries. Meanwhile, we use the initialized joints $J \in \mathbb{R}^{24\times 3}$, the perturbed joints $\hat{J} \in \mathbb{R}^{24\times 3}$ generated by the noise-augmented learning strategy, and the initialized sampled vertices $V \in \mathbb{R}^{N_v\times 3}$ as the corresponding positional inputs.

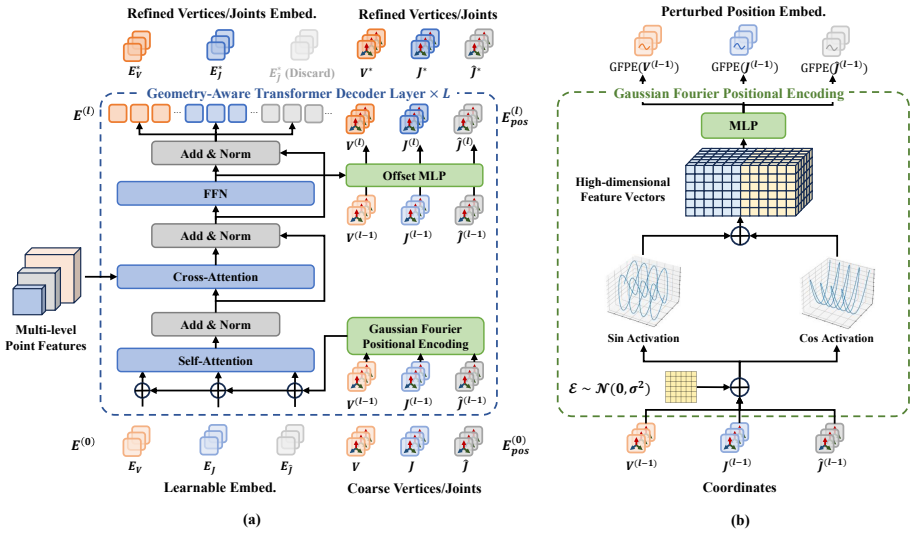

Fig. 3: Specific structure of the Geometry-Aware Transformer and Gaussian Fourier Positional Encoding module. **(a)** Geometry-Aware Transformer Decoder Layer. Vertex, joint, and noisy joint queries are jointly updated through self-attention and cross-attention over multi-level point features, and their refined coordinates are then obtained via an offset MLP. **(b)** Gaussian Fourier Positional Encoding module that projects 3D coordinates into high-dimensional positional encodings using random Gaussian projections with sine/cosine functions and an MLP.

Therefore, the initial queries and their corresponding positions are constructed by concatenating the embeddings or locations of vertex, joint, and noisy joint queries:

$$\begin{aligned} E^{(0)} &= \mathrm{Concat}(E_V, E_J, E_{\hat{J}}) \in \mathbb{R}^{N_q \times D}, \\ E^{(0)}_{\mathrm{pos}} &= \mathrm{Concat}(V, J, \hat{J}) \in \mathbb{R}^{N_q \times 3}, \end{aligned} \tag{2}$$

where $N_q = N_v + 24 + 24$ is determined by the numbers of vertices, joints, and noisy joints.

Geometry-Aware Transformer. Starting from the initialized joint and vertex queries described above, we employ a geometry-aware transformer to progressively refine them in a synergistic manner. As shown in Fig. 3(a), the transformer consists of L cascaded decoder layers. In each layer, self-attention models the synergistic interactions among joints and vertices, while cross-attention incorporates local geometric cues from multi-level point features to guide the refinement.

To retrieve local geometric details from sparse point clouds, we leverage the multi-level point features from the first three set abstraction (SA) layers of PointNet++, linearly project them to a unified dimension D, and concatenate them:

$$F_{\mathrm{pc}} = \mathrm{Concat}\big(W_1 F^{(1)},\ W_2 F^{(2)},\ W_3 F^{(3)}\big) \in \mathbb{R}^{N_s \times D}, \tag{3}$$

where W_i is the linear projection matrix for the i-th SA layer, $F^{(i)}$ denotes the point-cloud features from the i-th SA layer, and $N_s = \sum_{i=1}^{3} N_i$ is the total number of points after concatenation.

We further enhance the query positions at each decoder layer using a Gaussian Fourier Positional Encoding (GFPE) module, which injects high-frequency local geometric cues into the hybrid queries. At the l-th decoder layer, the hybrid query features are updated by applying self-attention and cross-attention:

$$\begin{aligned} F^{(l)}_{\text{joint}-\text{vertex}} &= E^{(l-1)} + \text{GFPE}(E^{(l-1)}_{\text{pos}}), \\ E^{(l)} &= \text{DecoderLayer}^{(l)}\left(F^{(l)}_{\text{joint}-\text{vertex}}; F_{\text{pc}}\right), \quad l = 1, \ldots, L, \end{aligned} \tag{4}$$

where $\text{DecoderLayer}^{(l)}(\cdot)$ denotes the l-th geometry-aware transformer decoder layer. Within each layer, self-attention captures the dependencies among joint, perturbed-joint, and vertex queries, enabling bidirectional information exchange between skeletal and mesh representations. Cross-attention then grounds these queries onto the local geometric observations by attending to F_{pc}, allowing sparse point-cloud geometry to guide the refinement process.

Finally, after feature refinement by the feed-forward network (FFN), the updated features are passed to an offset MLP to predict the 3D coordinate offsets $\Delta E^{(l)}_{\text{pos}}$, and the query positions are updated by:

$$E^{(l)}_{\text{pos}} = E^{(l-1)}_{\text{pos}} + \Delta E^{(l)}_{\text{pos}}. \tag{5}$$

After L layers of iterative refinement, the geometry-aware transformer outputs refined joints $J^* \in \mathbb{R}^{24\times 3}$ and refined vertices $V^* \in \mathbb{R}^{N_v \times 3}$. During training, V^* is supervised by a vertex loss. The refined joint positions J^*, together with the final geometry-aware joint features E^*_J, are then fed into the subsequent topology-aware mesh resolver.

Gaussian Fourier Positional Encoding Module. Inspired by [35], we note that directly using low-dimensional 3D coordinates or simple linear mappings makes it difficult for the attention mechanism to capture high-frequency local geometric details. Therefore, we introduce a Gaussian Fourier Positional Encoding (GFPE) module at the beginning of each decoder layer. Specifically, given an input coordinate $x \in \mathbb{R}^3$, the GFPE module projects it into a higher-dimensional feature space, and the positional encoding is defined as:

$$\gamma(x) = [\sin(2\pi x B),\ \cos(2\pi x B)], \tag{6}$$

where $B \in \mathbb{R}^{3\times(D/2)}$ is a projection matrix randomly sampled from a Gaussian distribution. We then use an MLP to map $\gamma(x)$ into a D-dimensional feature space that matches the model dimension, yielding the final positional encoding:

$$\text{GFPE}(x) = \text{MLP}(\gamma(x)) \in \mathbb{R}^D. \tag{7}$$

In addition to the query positions, GFPE is also applied to the 3D coordinates associated with the multi-level point features F_{pc}. By enriching both

query and point features with high-frequency positional cues, GFPE enables the transformer to better capture local geometric details and thus facilitates more accurate joint-mesh refinement.

3.3 Topology-Aware Mesh Resolver

To map the refined joint representations to a parametric human body model, we design a topology-aware mesh resolver, as illustrated in Fig. 2(c). Instead of predicting the mesh directly from independent joint features, a spatial GCN [17] is employed to explicitly model the anatomical topology of the human skeleton, enabling end-to-end learning of a direct mapping from the synergistically modeled features to joint rotations.

Feature aggregation. The input to the spatial GCN aggregates multi-scale information to ensure robust pose estimation and mesh reconstruction. Specifically, for each of the 24 joints, we concatenate three types of feature vectors: (1) the refined 3D joint coordinates $J^* \in \mathbb{R}^{24\times 3}$ produced by the geometry-aware transformer, providing explicit spatial guidance; (2) the refined geometry-aware joint features $E_J^* \in \mathbb{R}^{24\times D}$, capturing local surface details; and (3) the global feature $F_{\mathrm{g}} \in \mathbb{R}^{1024}$ extracted by the PointNet++ encoder, which is repeated to match the joint dimension. The resulting representation is:

$$F_{\mathrm{GCN}} = \mathrm{Concat}\big(J^*,\, E_J^*,\, \mathrm{Repeat}(F_{\mathrm{g}})\big) \in \mathbb{R}^{24\times(3+D+1024)}, \tag{8}$$

where $D = 256$. This aggregation effectively fuses local geometric joint details with the global structural context of the human body.

Skeleton-graph regression. To capture intrinsic spatial dependencies and anatomical constraints of the human body, we process F_{GCN} with a spatial GCN. Concretely, a skeleton graph is constructed, where nodes correspond to body joints and edges correspond to anatomical connections defined by the human structure. By propagating information along this skeletal topology, the GCN models correlations among neighboring joints, ensuring that the predicted pose is both structurally and biomechanically plausible. The GCN outputs a continuous 6D rotation representation [44] for each joint, which is subsequently converted into standard rotation matrices $\theta^* \in \mathbb{R}^{24\times 3\times 3}$.

SMPL mesh reconstruction. The predicted rotation matrices θ^* are then fed into a differentiable SMPL model [28]. Using Linear Blend Skinning (LBS), SMPL simultaneously outputs the dense 3D human mesh $V_{\mathrm{Mesh}} \in \mathbb{R}^{6890\times 3}$ and the 3D skeleton joints $J_{\mathrm{SMPL}} \in \mathbb{R}^{24\times 3}$.

3.4 Loss Functions

To achieve synergistic joint–mesh modeling and effectively constrain the skeletal structure while reconstructing fine surface details from sparse point clouds, we apply an L_2 loss to the refined vertices V^* output by the geometry-aware transformer. The vertex loss is defined as:

$$\mathcal{L}_{\mathrm{vertex}} = \sum_{k=1}^{N_v} \|V_k^* - V_{\mathrm{GT},k}\|_2^2\,, \tag{9}$$

where N_v is the number of sampled vertices, and V_{GT} denotes the ground-truth vertices indexed by I_{sample}.

Similarly, we apply L_2 supervision to the refined joints J^* from the Transformer, the final joints J_{SMPL} predicted by the SMPL model, and the pose parameters (rotation matrices) θ^* predicted by the spatial GCN. These losses are defined as:

$$\mathcal{L}_J = \sum_{j=1}^{24} \left\| J_j^* - J_{\mathrm{GT},j} \right\|_2^2, \tag{10}$$

$$\mathcal{L}_{J_{\mathrm{SMPL}}} = \sum_{j=1}^{24} \left\| J_{\mathrm{SMPL},j} - J_{\mathrm{GT},j} \right\|_2^2, \tag{11}$$

$$\mathcal{L}_\theta = \sum_{j=1}^{24} \left\| \theta_j^* - \theta_{\mathrm{GT},j} \right\|_F^2, \tag{12}$$

where J_{GT} and θ_{GT} represent the ground-truth joint coordinates and rotation matrices, respectively, and $\| \cdot \|_F$ denotes the Frobenius norm.

The overall training objective is the weighted sum of these loss terms:

$$\mathcal{L}_{\mathrm{total}} = \lambda_{\mathrm{vertex}} \mathcal{L}_{\mathrm{vertex}} + \lambda_J \mathcal{L}_J + \lambda_{J_{\mathrm{SMPL}}} \mathcal{L}_{J_{\mathrm{SMPL}}} + \lambda_\theta \mathcal{L}_\theta, \tag{13}$$

where $\lambda_{\mathrm{vertex}}, \lambda_J, \lambda_{J_{\mathrm{SMPL}}}$ and λ_θ are the corresponding loss balancing weights.

4 Experiments

4.1 Implementation Details

All experiments are conducted on a single NVIDIA RTX 3090 Ti GPU. The network is trained using the Adam optimizer [16] for 200 epochs with a batch size of 16. The initial learning rate is set to 1×10^{-4}. The weight decay coefficient is set to 1×10^{-4}. The dropout rate for the spatial GCN module is set to 0.5. Following our proposed synergistic joint-mesh modeling strategy, to ensure geometric consistency, the loss weights for the total loss function are configured as follows: the rotation matrix loss λ_θ is set to 1.0, while the joint losses λ_J, $\lambda_{J_{\mathrm{SMPL}}}$, and the vertex loss $\lambda_{\mathrm{vertex}}$ are all set to 5.0.

4.2 Datasets

LiDARHuman26M [21] is a large-scale outdoor multi-modal human motion capture dataset that synchronously collects LiDAR point clouds, RGB images, and ground-truth human motion information provided by IMUs. The dataset contains approximately 184K frames and about 26 million valid 3D points, covering a medium-to-long distance range of 12 m to 28 m. It is used to evaluate model performance on long-range human pose estimation from point clouds.
SLOPER4D [7] primarily targets global 4D human pose estimation in complex urban environments. It utilizes a LiDAR-camera system to collect human motion

Table 1: The proposed method was quantitatively evaluated against other methods on the LiDARHuman26M, SLOPER4D, and Human-M3 datasets. Best results are highlighted in bold. PA- stands for PA-MPJPE. LiDARCap† indicates single-frame input for LiDARCap.

Dataset	LiDARHuman26M					SLOPER4D					Human-M3				
Metrics	MPJPE	PA-	PVE	PCK3	PCK5	MPJPE	PA-	PVE	PCK3	PCK5	MPJPE	PA-	PVE	PCK3	PCK5
HybrIK [22]	105.17	92.51	155.61	79.08	92.23	55.73	43.66	87.63	92.93	97.81	85.61	66.80	121.58	84.05	93.26
P4T [10]	127.77	98.44	159.72	72.58	85.64	103.56	79.34	125.59	77.42	90.04	96.47	68.96	117.07	75.05	90.03
LiDARCap [21]	79.31	66.72	101.64	86.00	95.00	101.89	78.93	122.35	78.15	89.77	77.08	57.92	92.53	87.64	95.27
SAHSR [13]	107.40	88.16	134.69	77.41	89.88	72.86	52.51	84.78	89.24	97.13	105.31	70.82	134.86	78.16	90.38
VoteHMR [26]	133.48	108.74	164.72	70.52	84.89	54.02	41.51	63.51	93.90	98.15	104.52	70.09	127.04	78.56	90.57
LiDARCap† [21]	93.74	76.96	120.14	82.08	92.27	97.36	74.42	106.33	81.40	92.72	101.28	67.64	115.32	78.37	89.58
LiDAR-HMR [9]	76.07	67.14	101.73	86.12	94.75	47.78	36.30	49.75	94.90	98.35	76.35	56.53	86.92	87.63	94.78
SynHMR (Ours)	**74.25**	**65.04**	**96.22**	**87.26**	**95.39**	**34.09**	**26.87**	**41.32**	**97.03**	**99.02**	**56.76**	**43.03**	**70.70**	**93.10**	**97.15**

while leveraging inertial motion capture to obtain reliable ground-truth motion information. This dataset not only provides frame-wise 3D pose annotations but also includes SMPL parameters and scene point cloud information. Since the dataset does not provide an official training and testing split, we follow NE [41] for the train/test split.

Human-M3 [8] is a multi-view, multi-modal dataset designed for outdoor multi-person activities. It uses synchronized LiDAR point clouds and RGB videos for data collection, covers four typical scenes, and provides human pose keypoint annotations. The dataset contains approximately 89.6K human pose records for training and testing, involving a certain degree of natural occlusion.

4.3 Evaluation Metrics

We employ five metrics to evaluate model performance: MPJPE, PA-MPJPE, PVE, PCK-30, and PCK-50. **MPJPE**↓ denotes the mean per-joint position error (mm) between the predicted 3D joint positions and the ground truth. **PA-MPJPE**↓ first performs rigid registration between the predicted pose and the ground truth via Procrustes alignment, and then computes the mean joint error (mm) after alignment. **PVE (per-vertex error)**↓ measures the average Euclidean distance (mm) between each predicted mesh vertex and its corresponding ground-truth vertex. **PCK-30(PCK3)**↑/**PCK-50(PCK5)**↑ denote the percentage (%) of predicted joints whose distances to the corresponding ground-truth joints are within 0.3 and 0.5 times the torso length, respectively.

4.4 Comparison with the State-of-the-Art

Comparison methods. We compare our proposed approach with several state-of-the-art methods introduced in recent years. First, we evaluate against the image-based HMR method HybrIK [22], which analytically solves the swing rotation from 3D joints and regresses the 1-DoF twist angle to reconstruct the mesh. Next, we compare with temporal LiDAR-based HMR methods P4T [10]

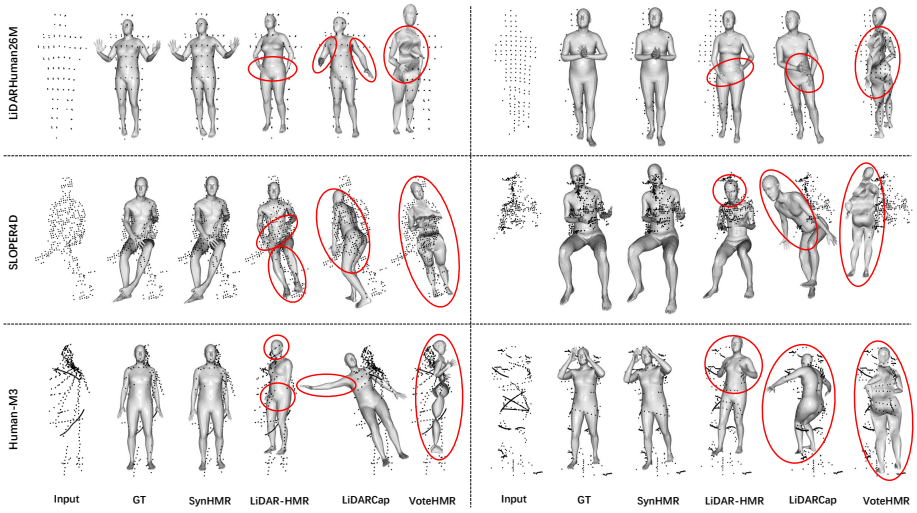

Fig. 4: Qualitative results on challenging samples. Our method can accurately predict joint positions in areas like limbs, where point clouds lack detailed information.

and LiDARCap [21]: P4T models spatiotemporal relations in point cloud sequences to capture dynamic motion, and LiDARCap is a classical method that follows a decoupled pipeline by first regressing 3D joints from temporal point cloud features and then fitting SMPL parameters through inverse kinematics and optimization. Finally, we compare with single-frame LiDAR-based HMR methods VoteHMR [26] and LiDAR-HMR [9]: VoteHMR regresses SMPL parameters from point cloud features via a voting mechanism; LiDAR-HMR first estimates a template 3D pose from the input point cloud and then progressively reconstructs the human mesh in a coarse-to-fine manner.

Quantitative comparison. As shown in Tab. 1, our method achieves best performance on all three datasets. Compared with the current state-of-the-art method LiDAR-HMR [9], our method reduces the MPJPE by 1.82 mm and the PVE by 5.51 mm on the LiDARHuman26M [21] dataset; reduces the MPJPE by 13.69 mm and the PVE by 8.43 mm on the SLOPER4D [7] dataset; and reduces the MPJPE by 19.59 mm and the PVE by 16.22 mm on the Human-M3 [8] dataset, respectively.

Qualitative comparison. Fig. 4 presents qualitative visualizations of our method. Across the three public datasets, our approach produces more accurate reconstructions than LiDAR-HMR [9], LiDARCap [21], and VoteHMR [26], especially on challenging examples. Since limbs are thinner than the torso, they contain fewer LiDAR points in the captured point clouds; our results demonstrate that the proposed method can localize limb joints more accurately under such sparse observations. These findings demonstrate that our synergistic joint-mesh modeling strategy effectively exploits the mutual reinforcement between skeletal joints

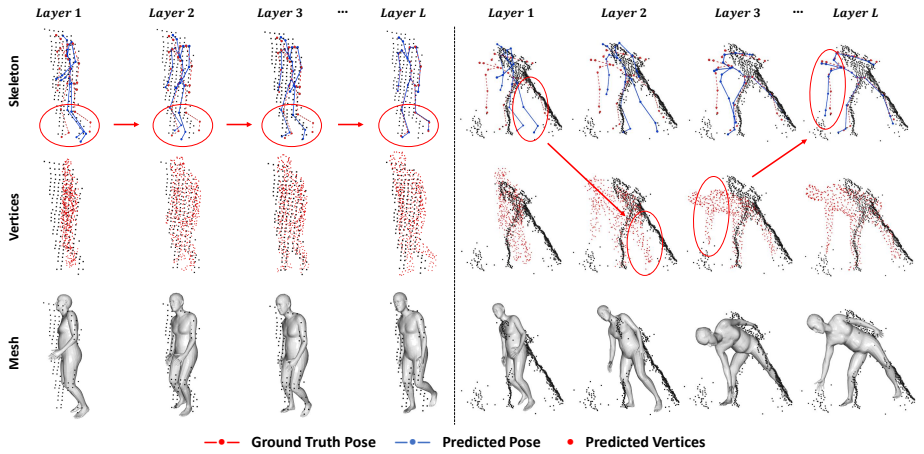

Fig. 5: Visualization of the progressive refinement process in the proposed synergistic joint-mesh modeling module. Horizontally, the initial pose and sampled vertices are progressively refined toward the ground truth across layers. Vertically, the skeleton guides the formation of the vertices, while the vertices in turn provide geometric constraints that help correct pose errors and reduce bias in the skeleton.

and mesh vertices, leading to more accurate limb recovery and more robust human mesh reconstruction under sparse point-cloud observations.

Layer-wise Visualization. Fig. 5 visualizes the effectiveness of the proposed synergistic joint-mesh modeling. The left part presents the layer-by-layer refinement process of skeleton joints, sampled vertices, and the reconstructed mesh, where the predictions are progressively improved and achieve a better final result. The right part visualizes the bidirectional interaction between skeleton and vertices across decoder layers. The skeleton provides a global structural prior that guides the formation and refinement of the sampled vertices toward anatomically plausible surfaces, even under sparse observations. As the vertices align with the observed surface, they serve as reliable anchors that impose explicit geometric constraints to correct pose errors and pull mislocalized joints back to plausible locations. Through this mutually reinforcing process, pose estimation and mesh reconstruction are progressively improved, producing results more faithful to the input point cloud. This figure demonstrates that the proposed synergistic joint-mesh modeling strategy substantially enhances reconstruction quality.

4.5 Ablation Study

Effect of synergistic components. In Tab. 2(a), we conduct an ablation study on the LiDARHuman26M dataset by progressively enabling each synergistic component to quantify its contribution. Specifically, row 1 reports the baseline without synergistic joint-mesh modeling, where all proposed modules are removed from SynHMR. This baseline uses PointNet++ for extracting point-

Table 2: Ablation study of the proposed method on the LiDARHuman26M dataset. In (a), Joint Queries and Mesh Queries denote two query branches in the Geometry-Aware Transformer, which together implement synergistic joint-mesh modeling. GFPE stands for Gaussian Fourier Positional Encoding, NAL stands for Noise-Augmented Learning, and LiDARCap† denotes LiDARCap with single-frame input. Best results are highlighted in bold.

(a) Ablation study of progressively enabling our proposed components

Joint Queries	Mesh Queries	GFPE	NAL	MPJPE	PVE
LiDARCap†		Baseline		93.74	120.14
✓				79.84	102.49
✓	✓			75.89	98.33
✓		✓		78.09	100.78
✓			✓	78.30	100.92
✓	✓	✓		75.62	98.12
✓	✓		✓	75.09	96.97
✓		✓	✓	76.89	99.60
✓	✓	✓	✓	**74.25**	**96.22**

(b) Ablation study of the number of sampled vertices

Vertices	MPJPE	PVE
32	75.21	97.38
64	74.92	97.07
128	74.75	96.72
256	74.25	96.22
512	**74.21**	**96.00**

(c) Ablation study of the number of Geometry -Aware Transformer layers

L	3	4	5	**6**	7
MPJPE	75.82	74.98	74.57	**74.25**	74.28
PVE	98.18	97.02	96.37	**96.22**	96.34

cloud features, an MLP decoder for predicting 3D joints, and a GCN for SMPL parameter regression, which is equivalent to LiDARCap [21] with single-frame input. Compared with using only joint queries, adding mesh queries improves MPJPE/PVE from 79.84/102.49 to 75.89/98.33, suggesting that the gain mainly comes from synergistic joint-mesh modeling rather than from transformer capacity, cross-attention to point-cloud features, NAL, or auxiliary vertex supervision. GFPE and NAL further improve performance by enhancing geometric positional encoding and training robustness, respectively. The full model achieves the best performance, reducing MPJPE and PVE by 19.49 mm and 23.92 mm over the baseline.

Number of sampled vertices. We investigate how the number of sampled mesh vertices affects the performance of synergistic joint-mesh modeling. As shown in Tab. 2(b), increasing the number of sampled vertices from 32 to 512 consistently improves reconstruction accuracy: MPJPE decreases from 75.21 mm to 74.21 mm, and PVE decreases from 97.38 mm to 96.00 mm. This suggests that denser vertex sampling provides richer geometric supervision and strengthens the synergy between joints and mesh. However, it also increases computational cost, leading to lower efficiency. In practice, we adopt 256 sampled vertices, as it offers

Table 3: Efficiency comparison with existing LiDAR-based human mesh reconstruction methods on LiDARHuman26M. Latency(ms), GFLOPs, Params(M), MPJPE, and PVE are reported, where lower values indicate better performance.

Method	Latency	GFLOPs	Params	MPJPE	PVE
VoteHMR [26]	18.88	4.35	**1.78**	133.48	164.72
LiDAR-HMR [9]	55.28	13.94	46.40	76.07	101.73
LiDARCap [21]	8.62	9.18	34.93	79.31	101.64
SynHMR (Ours)	**7.56**	**3.87**	14.80	**74.25**	**96.22**

a favorable trade-off between accuracy and efficiency, achieving near-saturated performance while keeping runtime overhead within a manageable range.

Number of Geometry-Aware Transformer Layers. The geometry-aware transformer is used to refine pose estimation and mesh reconstruction. Stacking more layers can improve performance, as shown in Tab. 2(c). This demonstrates the effectiveness of our proposed geometry-aware transformer's progressive refinement strategy. However, as the number of layers becomes sufficiently large, the advantage of increasing the number of layers gradually diminishes, meaning the model has reached its performance ceiling.

4.6 Efficiency Analysis

We report the efficiency comparisons in Tab. 3. SynHMR achieves the best reconstruction accuracy with the lowest latency and GFLOPs. This efficiency benefits from its single-frame design, sparse query refinement, and training-only NAL, avoiding temporal modeling [21], dense upsampling [9], and point-wise voting [26]. SynHMR achieves a favorable balance between accuracy and efficiency.

5 Conclusion

In this paper, we present SynHMR for single-frame LiDAR-based human mesh reconstruction. By jointly modeling skeletal joints and sampled mesh vertices, SynHMR enables pose and mesh to refine each other under sparse point-cloud observations. A Geometry-Aware Transformer and a Noise-Augmented Learning strategy further improve bidirectional refinement and robustness. Experiments on three public datasets demonstrate that SynHMR achieves state-of-the-art performance, highlighting the potential of synergistic joint-mesh modeling for reliable LiDAR-based human understanding.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (International Collaboration Special Project, No. SQ2023YFE0102775), and the NSF of China (No. 62572242).

References

1. An, X., Zhao, L., Gong, C., Li, J., Yang, J.: Pre-training a density-aware pose transformer for robust lidar-based 3d human pose estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 1755–1763 (2025)
2. Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J.: Scape: shape completion and animation of people. In: ACM Siggraph 2005 Papers, pp. 408–416 (2005)
3. Bashirov, R., Ianina, A., Iskakov, K., Kononenko, Y., Strizhkova, V., Lempitsky, V., Vakhitov, A.: Real-time rgbd-based extended body pose estimation. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 2807–2816 (2021)
4. Bright, J., Balaji, B., Prakash, H., Chen, Y., Clausi, D.A., Zelek, J.: Distribution and depth-aware transformers for 3d human mesh recovery. arXiv preprint arXiv:2403.09063 **4**(6), 7 (2024)
5. Choi, H., Moon, G., Lee, K.M.: Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In: European Conference on Computer Vision. pp. 769–787. Springer (2020)
6. Cong, P., Xu, Y., Ren, Y., Zhang, J., Xu, L., Wang, J., Yu, J., Ma, Y.: Weakly supervised 3d multi-person pose estimation for large-scale scenes based on monocular camera and single lidar. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 461–469 (2023)
7. Dai, Y., Lin, Y., Lin, X., Wen, C., Xu, L., Yi, H., Shen, S., Ma, Y., Wang, C.: Sloper4d: A scene-aware dataset for global 4d human pose estimation in urban environments. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 682–692 (2023)
8. Fan, B., Wang, S., Guo, W., Zheng, W., Feng, J., Zhou, J.: Human-m3: A multi-view multi-modal dataset for 3d human pose estimation in outdoor scenes. arXiv preprint arXiv:2308.00628 (2023)
9. Fan, B., Zheng, W., Feng, J., Zhou, J.: Lidar-hmr: 3d human mesh recovery from lidar. IEEE Transactions on Multimedia (2025)
10. Fan, H., Yang, Y., Kankanhalli, M.: Point 4d transformer networks for spatio-temporal modeling in point cloud videos. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14204–14213 (2021)
11. Fang, X., Yang, J., Rao, J., Wang, L., Deng, Z.: Single rgb-d fitting: Total human modeling with an rgb-d shot. In: Proceedings of the 25th ACM Symposium on Virtual Reality Software and Technology. pp. 1–11 (2019)
12. Hinojosa, C., Niebles, J.C., Arguello, H.: Learning privacy-preserving optics for human pose estimation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2573–2582 (2021)
13. Jiang, H., Cai, J., Zheng, J.: Skeleton-aware 3d human shape reconstruction from point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5431–5441 (2019)
14. Kamel, A., Sheng, B., Li, P., Kim, J., Feng, D.D.: Hybrid refinement-correction heatmaps for human pose estimation. IEEE Transactions on Multimedia **23**, 1330–1342 (2020)
15. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7122–7131 (2018)

16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
17. Kipf, T.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
18. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: 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. pp. 2252–2261 (2019)
19. Kolotouros, N., Pavlakos, G., Daniilidis, K.: Convolutional mesh regression for single-image human shape reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4501–4510 (2019)
20. Kovács, L., Bódis, B.M., Benedek, C.: Lidpose: real-time 3d human pose estimation in sparse lidar point clouds with non-repetitive circular scanning pattern. Sensors **24**(11), 3427 (2024)
21. Li, J., Zhang, J., Wang, Z., Shen, S., Wen, C., Ma, Y., Xu, L., Yu, J., Wang, C.: Lidarcap: Long-range marker-less 3d human motion capture with lidar point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20502–20512 (2022)
22. Li, J., Xu, C., Chen, Z., Bian, S., Yang, L., Lu, C.: Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3383–3393 (2021)
23. Lian, J., Wang, D., Zhu, S., Wu, Y., Li, C.: Transformer-based attention network for vehicle re-identification. Electronics **11**(7), 1016 (2022)
24. Lin, K., Wang, L., Liu, Z.: End-to-end human pose and mesh reconstruction with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1954–1963 (2021)
25. Lin, K., Wang, L., Liu, Z.: Mesh graphormer. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12939–12948 (2021)
26. Liu, G., Rong, Y., Sheng, L.: Votehmr: Occlusion-aware voting network for robust 3d human mesh recovery from partial point clouds. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 955–964 (2021)
27. Liu, Y., Qiu, C., Zhang, Z.: Deep learning for 3d human pose estimation and mesh recovery: A survey. Neurocomputing **596**, 128049 (2024)
28. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. In: Seminal Graphics Papers: Pushing the Boundaries, Volume 2, pp. 851–866 (2023)
29. Mao, W., Ge, Y., Shen, C., Tian, Z., Wang, X., Wang, Z., den Hengel, A.v.: Poseur: Direct human pose regression with transformers. In: European conference on computer vision. pp. 72–88. Springer (2022)
30. Osman, A.A., Bolkart, T., Black, M.J.: Star: Sparse trained articulated human body regressor. In: European Conference on Computer Vision. pp. 598–613. Springer (2020)
31. Pan, Z., Zhong, Z., Guo, W., Chen, Y., Feng, J., Zhou, J.: Licampose: combining multi-view lidar and rgb cameras for robust single-timestamp 3d human pose estimation. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 2484–2494. IEEE (2025)
32. Peng, X.B., Kanazawa, A., Toyer, S., Abbeel, P., Levine, S.: Variational discriminator bottleneck: Improving imitation learning, inverse rl, and gans by constraining information flow. arXiv preprint arXiv:1810.00821 (2018)

33. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
34. Ren, Y., Han, X., Yao, Y., Long, X., Sun, Y., Ma, Y.: Livehps++: Robust and coherent motion capture in dynamic free environment. In: European Conference on Computer Vision. pp. 127–144. Springer (2024)
35. Tancik, M., Srinivasan, P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J., Ng, R.: Fourier features let networks learn high frequency functions in low dimensional domains. Advances in neural information processing systems **33**, 7537–7547 (2020)
36. Tian, Y., Zhang, H., Liu, Y., Wang, L.: Recovering 3d human mesh from monocular images: A survey. IEEE transactions on pattern analysis and machine intelligence **45**(12), 15406–15425 (2023)
37. Wang, J., Yan, S., Dai, B., Lin, D.: Scene-aware generative network for human motion synthesis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12206–12215 (2021)
38. Wang, K., Xie, J., Zhang, G., Liu, L., Yang, J.: Sequential 3d human pose and shape estimation from point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7275–7284 (2020)
39. Wu, X., Zhang, H., Kong, C., Wang, Y., Ju, Y., Zhao, C.: Lidar-based 3-d human pose estimation and action recognition for medical scenes. IEEE Sensors Journal **24**(9), 15531–15539 (2024)
40. Ye, D., Xie, Y., Chen, W., Zhou, Z., Ge, L., Foroosh, H.: Lpformer: Lidar pose estimation transformer with multi-task network. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 16432–16438. IEEE (2024)
41. Zhang, J., Mao, Q., Hu, G., Shen, S., Wang, C.: Neighborhood-enhanced 3d human pose estimation with monocular lidar in long-range outdoor scenes. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 7169–7177 (2024)
42. Zheng, J., Shi, X., Gorban, A., Mao, J., Song, Y., Qi, C.R., Liu, T., Chari, V., Cornman, A., Zhou, Y., et al.: Multi-modal 3d human pose estimation with 2d weak supervision in autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4478–4487 (2022)
43. Zheng, Z., Yu, T., Wei, Y., Dai, Q., Liu, Y.: Deephuman: 3d human reconstruction from a single image. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7739–7749 (2019)
44. Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5745–5753 (2019)