Check for updates

RESEARCH ARTICLE

Amortized inverse kinematics via graph attention for real-time human avatar animation

[version 1; peer review: 1 approved with reservations]

Muhammad Saif Ullah Khan[1,2], Chen-Yu Wang[1], Tim Prokosch[2], Michael Lorenz[1], Bertram Taetz[3], Didier Stricker[1,2]

[1]Augmented Vision, German Research Center for Artificial Intelligence, Kaiserslautern, Rhineland-Palatinate, 67663, Germany
[2]Rheinland-Pfalzische Technische Universitat Kaiserslautern-Landau, Kaiserslautern, Rhineland-Palatinate, 67663, Germany
[3]International University of Applied Sciences, Erfurt, Thuringia, 99084, Germany

v1 **First published:** 29 Aug 2026, **6**:314
https://doi.org/10.12688/openreseurope.24416.1
Latest published: 29 Aug 2026, **6**:314
https://doi.org/10.12688/openreseurope.24416.1

Abstract

Inverse kinematics (IK) is a core operation in animation, robotics, and biomechanics: given Cartesian constraints, recover joint rotations under a known kinematic tree. In many real-time human avatar pipelines, the available signal per frame is a sparse set of tracked 3D joint positions, whereas animation systems require *joint orientations* to drive skinning. Recovering full orientations from positions is underconstrained, most notably because twist about bone axes is ambiguous, and classical IK solvers typically rely on iterative optimization that can be slow and sensitive to noisy inputs. We introduce **IK-GAT** , a lightweight graph-attention network that reconstructs full-body joint orientations from 3D joint positions in a single forward pass. The model performs message passing over the skeletal parent-child graph to exploit kinematic structure during rotation inference. To simplify learning, IK-GAT predicts rotations in a *bone-aligned world-frame* representation anchored to rest-pose bone frames. This parameterization makes the twist axis explicit and is *exactly invertible* to standard parent-relative local rotations given the kinematic tree and rest pose. The network uses a continuous 6D rotation representation and is trained with a geodesic loss on SO(3) together with an optional forward-kinematics consistency regularizer. IK-GAT produces animation-ready local rotations that can directly drive a rigged avatar or be converted to pose parameters of SMPL-like body models for real-time and online applications. With 374 K parameters and over 650 FPS on CPU, IK-GAT outperforms VPoser-based per-frame iterative optimization without warm-start at significantly lower cost, and is robust to initial pose and input noise.

Open Peer Review

Approval Status ?

	1
version 1 29 Aug 2026	? view

1. **Yushan Pan**, Xi'an Jiaotong Liverpool University, Suzhou, China

Any reports and responses or comments on the article can be found at the end of the article.

Keywords
Human Motion Tracking, Avatar Animation, Graph Neural Networks, Neural Inverse Kinematics

This article is included in the Horizon Europe gateway.

Corresponding author: Muhammad Saif Ullah Khan (muhammad_saif_ullah.khan@dfki.de)

Author roles: Khan MSU: Conceptualization, Data Curation, Investigation, Methodology, Software, Visualization, Writing – Original Draft Preparation; **Wang CY**: Conceptualization, Data Curation, Methodology, Software, Writing – Original Draft Preparation; **Prokosch T**: Validation, Writing – Original Draft Preparation; **Lorenz M**: Conceptualization, Project Administration, Supervision, Writing – Review & Editing; **Taetz B**: Validation, Writing – Original Draft Preparation, Writing – Review & Editing; **Stricker D**: Funding Acquisition, Project Administration, Resources

Competing interests: No competing interests were disclosed.

Grant information: This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No [101135724] (LUMINOUS); [101298864] VICTOR-XR; addressing Topic [HORIZON-CL4-2023-HUMAN-01-21].
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2026 Khan MSU *et al*. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Khan MSU, Wang CY, Prokosch T *et al.* **Amortized inverse kinematics via graph attention for real-time human avatar animation [version 1; peer review: 1 approved with reservations]** Open Research Europe 2026, **6**:314 https://doi.org/10.12688/openreseurope.24416.1

First published: 29 Aug 2026, **6**:314 https://doi.org/10.12688/openreseurope.24416.1

1. Introduction

Human avatar animation in production systems, such as films, gaming and VR, often uses an offline pipeline where motion capture and animation stages are distinct and asynchronous.[1] Motion data is first captured—traditionally using marker-based systems[2]—and later refined through post-processing stages such as retargeting,[3] filtering, and manual adjustment before being applied to animated avatars. These steps ensure that the captured motion translates into visually coherent joint rotations suitable for skinning and deformation.[4,5] In recent years, however, motion capture has increasingly relied on sparse and markerless sensing modalities, including vision-based pose estimators and VR/AR tracking devices.[6,7] Such systems typically provide only 3D joint positions rather than full skeletal orientations,[8] yet high-quality animation still requires *joint rotations*: rotations determine skinning, preserve limb twist, and control visually salient deformations in shoulders, forearms, and thighs.[4,9,5] Recovering these orientations from positions is fundamentally underconstrained—multiple rotation configurations can yield nearly identical joint locations, with twist about bone-aligned axes being particularly ambiguous.[10,11,6]

Classical inverse-kinematics (IK) solvers address this ambiguity by introducing additional structure and constraints. Rather than inferring rotations purely from joint positions, they formulate the problem as a constrained optimization over the kinematic chain, incorporating anatomical joint limits, preferred postures, damping terms, and task-specific objectives.[12,11] Common methods include Jacobian-based solvers, cyclic coordinate descent, and FABRIK,[12,13,14] which iteratively adjust joint rotations so that the resulting forward kinematics match the observed joint positions. While effective in controlled settings, such optimization-based methods present practical drawbacks for real-time full-body use: per-frame iteration increases latency and can amplify positional noise into rotational jitter; convergence depends on carefully tuned step sizes and damping parameters; and incorporating many joint constraints often requires manual specification and domain-specific heuristics.[12,14,11,15] When the objective is *full-body* orientation recovery[10,16] rather than end-effector placement, the solver must reconcile a large set of noisy positional constraints with kinematic structure and plausible articulation, making the optimization expensive and brittle.[11]

In this work, we instead learn an *amortized inverse-kinematics mapping* that directly predicts joint orientations from 3D joint positions in a single forward pass. As illustrated in Figure 1, the proposed IK-GAT model operates as a lightweight inference module between upstream tracking systems and downstream animation pipelines. Given tracked 3D joint positions, the network predicts per-joint rotations in a continuous 6D representation.[17] These predictions are converted to SO(3) using Gram–Schmidt orthogonalization,[17] yielding canonical *bone-aligned world rotations* whose primary axis follows the bone direction. Using precomputed rest-pose bone frames, an analytic mapping then recovers rig-specific local rotations required by standard animation rigs.

This decomposition separates learned and deterministic components of the problem. The network focuses on predicting a geometrically consistent world-frame orientation for each bone, while kinematic composition and localization are recovered exactly through the known skeletal structure. The resulting system acts as a lightweight neural inverse-kinematics module[18] that can be integrated into multiple animation pipelines. For example, predicted joint orientations can drive SMPL-like[19] parametric body models within existing workflows, or they can be streamed directly to a target character rig or game engine without relying on an intermediate body model, enabling real-time animation from joint positions only.

Contributions. This paper makes the following contributions:

- We introduce **IK-GAT**, a lightweight graph-attention network that predicts physically-plausible per-joint orientations from 3D joint positions at high framerates, enabling real-time animation.
- We propose using **bone-aligned rotations in world coordinates** for a more stable learning objective while remaining *exactly invertible* to rig-specific world-frames and standard parent-relative local rotations given a known rest pose and a kinematic tree.
- We incorporate an explicit **kinematic prior** by using the skeletal parent–child graph as the message-passing topology, enabling stable and efficient learning of articulated motion.
- We evaluate across heterogeneous motion sources for SMPL and Unreal Engine (UE) avatar animation, and analyze robustness under noisy joint inputs.

A preprint version of this work[20] was submitted on arXiv.

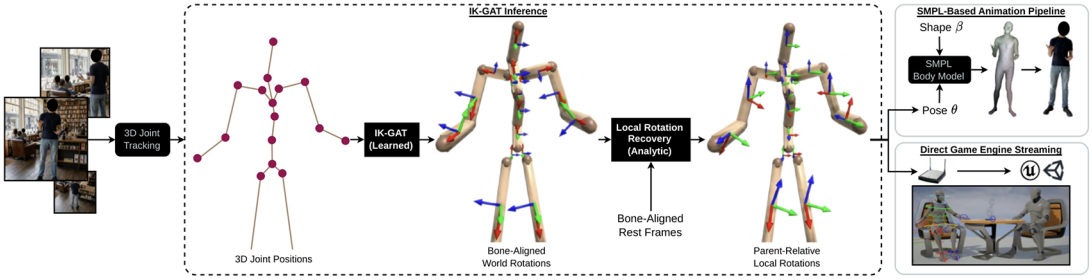

Figure 1. Amortized inverse kinematics. Given 3D joint positions, IK-GAT predicts per-joint rotations in bone-aligned world space and analytically recovers standard local rotations, enabling direct animation or use in body models.

2. Related work

We review prior work on IK, optimization-based body model fitting, learning-based IK, and skeleton-aware neural architectures, and position our setting relative to Human Mesh Recovery (HMR).

Inverse kinematics in robotics and animation. Inverse kinematics seeks joint parameters that satisfy Cartesian constraints under a known kinematic chain. Closed-form analytic solutions exist only for restricted mechanisms with special structure, such as those satisfying Pieper's conditions.[21] For general articulated figures, most systems rely on iterative numerical optimization. Common approaches include Jacobian-based solvers,[12] cyclic coordinate descent (CCD),[13] and FABRIK.[14] In biomechanics analysis, OpenSim[22] formulates IK as a constrained least-squares problem that matches observed markers or joints, but this only works reliably when enough markers to recover an observable orientation per segment or joint are available. This is different to IK from joints only which usually does not allow for observable internal rotation estimation. While widely used, iterative solvers are sensitive to observation noise and solver hyperparameters, and their per-frame optimization cost can become prohibitive when many joints and constraints must be satisfied in real time.[23]

Optimization-based body model fitting. Parametric human body models such as SMPL and SMPL-X provide a standardized pose representation and differentiable forward kinematics for joints and mesh vertices.[19,24] Optimization-based fitting methods such as SMPLify recover pose parameters from 2D or 3D observations using priors and regularization.[25] Learned pose priors (e.g., VPoser) further constrain solutions toward plausible human configurations.[26] Although effective for offline reconstruction, these approaches remain per-instance optimization procedures and can introduce latency and temporal instability when applied to real-time animation.

Human mesh recovery. Human Mesh Recovery (HMR) methods regress body pose and shape directly from images or keypoints. Early work introduced end-to-end regression of SMPL parameters from monocular images,[27] and later systems improved robustness and accuracy through stronger architectures and temporal reasoning.[28,29] Unlike HMR, our setting assumes that 3D joint positions are already available from an upstream tracking system and focuses exclusively on recovering stable joint orientations suitable for animation pipelines.

Learning-based and hybrid inverse kinematics. Learning-based IK amortizes the mapping from Cartesian constraints to joint parameters using neural networks. Several works combine learned inference with analytic structure. HybrIK recovers rotations from RGB images using a hybrid analytic–neural formulation with explicit twist handling.[10] HierarIK exploits the kinematic hierarchy to improve robustness under noisy keypoints,[8] while MANIKIN and related systems target sparse tracking scenarios and incorporate biomechanical constraints to produce plausible full-body motion.[15] Other sparse motion tracking methods like Deep Inertial Poser[30] and TransPose[31] further demonstrate the feasibility of learning-based IK under limited observations and using IMU data. Our work follows this direction but focuses specifically on only dense 3D joint inputs and real-time single-frame rotation recovery for character animation. **Skeleton-aware learning.** Human skeletons naturally form graphs defined by anatomical connectivity. Graph neural networks have therefore become a common architecture for skeleton-based reasoning. ST-GCN introduced spatio-temporal graph convolutions for skeleton sequences,[32] while SemGCN learned semantic edge weights for pose regression.[33] Graph Attention Networks further introduced attention-based message passing over graph neighborhoods.[34] These architectures demonstrate the benefit of explicitly modeling the kinematic tree structure, motivating our use of graph attention over skeletal connectivity.

Rotation representations. Learning rotations requires representations compatible with Euclidean neural networks while respecting SO(3) geometry. The continuous 6D representation predicts two rotation-matrix columns followed by orthonormalization, avoiding discontinuities present in Euler angles and quaternions.[17] Rotation errors are commonly measured using geodesic distances on SO(3).[35] Our model adopts these standard choices while introducing a bone-aligned world-frame parameterization.

3. Method

3.1 Problem formulation

We consider a temporally causal stream of 3D joint positions expressed in root-centered coordinates (root space), observed at time t as a rooted kinematic tree with pose

$$\tilde{\mathbf{J}}_t = \left\{ \tilde{\mathbf{j}}_t^{(i)} \in \mathbb{R}^3 \right\}_{i=0}^{N-1}, \tag{1}$$

where $\tilde{\mathbf{j}}_t^{(0)} = \mathbf{0}$ denotes the root joint, N is the number of joints, i is the joint index, and joints are sorted in a parent-first order. Let $\pi(i) \in [-1, N-1]$ denote the parent of joint i, with $\pi(0) = -1$ for the root. The k-th ancestor of joint i is defined recursively as

$$\pi^0(i) = i, \pi^k(i) = \pi\left(\pi^{k-1}(i)\right), k \geq 1,$$

and for each joint i, there exists a finite depth $d(i) = K$ such that $\pi^K(i) = 0$ and $\pi^{K+1}(i) = -1$; equivalently, repeated application of π reaches the root in finitely many steps.

To support real-time, online, and memory-efficient avatar animation, we aim to recover parent-relative local joint rotations suitable for standard animation pipelines, in a single per-frame forward pass of a lightweight neural network without iterative optimizations, access to future poses, or an unbounded access to past poses.

3.1.1 Forward kinematics (FK)

Let $\mathbf{R}_{t,\text{loc}}^{(i)} \in \text{SO}(3)$ denote the parent-relative local rotation of joint i at time t, and define $\mathbf{R}_{t,\text{w}}^{(0)} = \mathbf{R}_{t,\text{loc}}^{(0)}$ as the world rotation of the root. World-space joint rotation matrices are then obtained recursively as

$$\mathbf{R}_{t,\text{w}}^{(i)} = \begin{cases} \mathbf{R}_{t,\text{loc}}^{(0)}, & i = 0, \\ \mathbf{R}_{t,\text{w}}^{(\pi(i))} \mathbf{R}_{t,\text{loc}}^{(i)} & \text{otherwise.} \end{cases} \tag{2}$$

Let $\overline{\mathbf{J}} = \left\{ \overline{\mathbf{j}}^{(i)} \in \mathbb{R}^3 \right\}_{i=0}^{N-1}$ be the unposed, template rig with joints at rest, $\overline{\mathbf{d}}^{(i)} = \overline{\mathbf{j}}^{(i)} - \overline{\mathbf{j}}^{(\pi(i))}$ be the fixed rest-pose offset from the parent to joint $i \in (0, N)$, and $\overline{\mathbf{d}}^{(0)} = 0$ for the root, and $\overline{\mathbf{d}}^{(i)} \neq \mathbf{0}$ for $i > 0$. Root-space joint positions are defined recursively as

$$\tilde{\mathbf{j}}_t^{(i)} = \begin{cases} \mathbf{0}, & i = 0, \\ \tilde{\mathbf{j}}_t^{(\pi(i))} + \mathbf{R}_{t,\text{w}}^{(\pi(i))} \overline{\mathbf{d}}^{(i)}, & \text{otherwise.} \end{cases} \tag{3}$$

We denote this mapping by $\text{FK}\left(\left\{ \mathbf{R}_{t,\text{loc}}^{(i)} \right\}_{i=0}^{N-1} \right)$.

3.2 Inverse kinematics via bone-aligned world frames

Inverse kinematics (IK) is the inverse task of recovering a set of local rotations given 3D joints, such that $\text{FK}\left(\left\{ \mathbf{R}_{t,\text{loc}}^{(i)} \right\}_{i=0}^{N-1} \right) \approx \tilde{\mathbf{J}}_t$. This is ill-posed: joint positions underconstrain rotations, most notably through twist around bone-aligned axes and in the presence of noisy or missing joints, as shown in Figure 2.

Key idea. We separate the IK problem into a learned component and a deterministic component. Rather than regressing local rotations directly, we learn an intermediate representation that (i) is stable for regression and (ii) admits an exact, closed-form conversion to standard local rotations. The network predicts per-joint rotation matrices representing *bone-aligned world frames*, while the conversion to animation-compatible parent-relative local rotations is analytic and lossless. This reduces the complexity of what must be learned: the model is not required to learn kinematic composition rules or rest-pose basis conventions, only the pose-dependent orientations in a canonicalized frame.

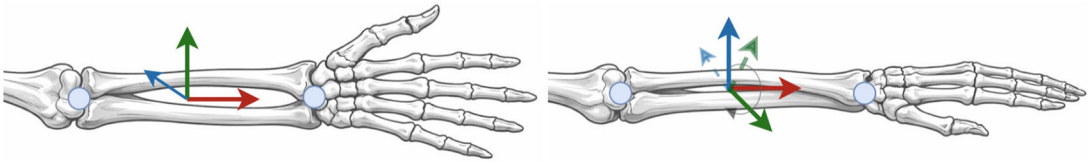

Figure 2. Twist ambiguity. Identical joint positions (blue) can be produced by any rotation about the bone axis (red), making inverse kinematics ill-posed without strong anatomical priors. Neural IK methods therefore learn these priors from data.

The bone-aligned world rotation matrices differ from the standard world-space rotation matrices in how the underlying local coordinate frames are defined. In typical rigged avatars, each joint is associated with an author-defined local frame whose orientation relative to the bone is arbitrary and varies across rigs. Consequently, the corresponding world rotation matrices depend on these rig-specific conventions. In contrast, we define a canonical local frame at each joint directly from the rest-pose geometry. This yields a consistent, bone-aligned frame whose x-axis follows the bone direction and whose remaining axes are fixed deterministically.

3.2.1 Rest-pose bone frames (precomputed once per rig)

Let $\mathbf{u}$ be a fixed unit vector denoting the global up direction. For each joint i in a given rig with rest-pose $\bar{\mathbf{J}}$, we select a *primary child* $c(i)$ using a deterministic choice among children. For joints with multiple children, we prefer the child whose offset aligns most with global up, and break ties by longer bone length.

We define a directed edge $(s(i), t(i))$ for each joint i as

$$(s(i),t(i)) = \begin{cases} (i,c(i)), & \text{if } i \text{ has at least one child,} \\ (\pi(i),i), & \text{otherwise,} \end{cases} \tag{4}$$

where $\pi(i)$ denotes the parent of i. Assuming nonzero bone lengths, the bone direction at rest is then defined as

$$\bar{\mathbf{x}}^{(i)} = \frac{\bar{\mathbf{j}}^{(t(i))} - \bar{\mathbf{j}}^{(s(i))}}{\|\bar{\mathbf{j}}^{(t(i))} - \bar{\mathbf{j}}^{(s(i))}\|_2}. \tag{5}$$

To fix the remaining degree of freedom (twist), we choose a reference direction

$$\mathbf{r}^{(i)} = \begin{cases} \mathbf{u}, & i = 0, \\ \bar{\mathbf{y}}^{(\pi(i))}, & \text{otherwise,} \end{cases} \tag{6}$$

where $\bar{\mathbf{y}}^{(\pi(i))}$ is the parent's propagated reference y-axis, recursively computed as defined in (7). We then construct an orthonormal, right-handed basis via Gram-Schmidt[17]:

$$\tilde{\mathbf{y}}^{(i)} = \mathbf{r}^{(i)} - \langle \mathbf{r}^{(i)}, \bar{\mathbf{x}}^{(i)} \rangle \bar{\mathbf{x}}^{(i)}, \bar{\mathbf{y}}^{(i)} = \frac{\tilde{\mathbf{y}}^{(i)}}{\|\tilde{\mathbf{y}}^{(i)}\|_2}, \tag{7}$$

$$\bar{\mathbf{z}}^{(i)} = \bar{\mathbf{x}}^{(i)} \times \bar{\mathbf{y}}^{(i)}. \tag{8}$$

In the degenerate case where $\mathbf{r}^{(i)}$ is (near-)collinear with $\bar{\mathbf{x}}^{(i)}$, the projection in (7) yields $\|\tilde{\mathbf{y}}^{(i)}\|_2 \approx 0$, making the normalization ill-defined. In this case, we detect the degeneracy and reapply the construction using a fallback reference direction $\mathbf{r}_{\text{fb}}$, chosen as the global up vector $\mathbf{u}$. The resulting frame is well-defined provided at least one of $\mathbf{r}^{(i)}$ or $\mathbf{r}_{\text{fb}}$ is not collinear with $\bar{\mathbf{x}}^{(i)}$.

In practice, this condition is easily satisfied for human-like kinematic trees in a rest pose: bone directions are diverse and rarely align with a single global axis across parent-child chains, ensuring that either the propagated parent reference or the global up vector provides a valid, non-collinear direction.

The resulting *bone-aligned rest frame* is represented by

$$\mathbf{B}_{\text{rest}}^{(i)} = \left[\bar{\mathbf{x}}^{(i)}\ \bar{\mathbf{y}}^{(i)}\ \bar{\mathbf{z}}^{(i)}\right] \in \mathrm{SO}(3), \tag{9}$$

which is the rotation matrix mapping the bone-local rest frame to world coordinates at rest. This computation is performed once per target rig and does not depend on the observed motion.

3.2.2 Bone-aligned world frame representations

Let $\mathbf{R}_{t,\text{w}}^{(i)}$ be the joint world rotation under FK. We define the rotation matrix representing the *bone-aligned world frame*

$$\mathbf{R}_{t,\text{bone}}^{(i)} = \mathbf{R}_{t,\text{w}}^{(i)} \mathbf{B}_{\text{rest}}^{(i)}. \tag{10}$$

Intuitively, the matrix $\mathbf{R}_{t,\text{bone}}^{(i)}$ maps the canonical bone-aligned local frame to world coordinates. By construction, the local x-axis of this frame is defined from the rest-pose bone direction, and under $\mathbf{R}_{t,\text{bone}}^{(i)}$ it is transformed to the current world-space direction of the rest-defined local x-axis. This makes the twist axis (rotation about the bone direction) explicit while preserving all three rotational DoFs.

3.2.3 Exact invertibility to parent-relative local rotations

Given $\mathbf{R}_{t,\text{bone}}^{(i)}$ and $\mathbf{B}_{\text{rest}}^{(i)}$, we recover world rotations exactly via

$$\mathbf{R}_{t,\text{w}}^{(i)} = \mathbf{R}_{t,\text{bone}}^{(i)} \left(\mathbf{B}_{\text{rest}}^{(i)}\right)^{\top}, \tag{11}$$

and then recover local rotations by the standard parent-relative relation

$$\mathbf{R}_{t,\text{loc}}^{(i)} = \begin{cases} \mathbf{R}_{t,\text{w}}^{(i)}, & \pi(i) = -1, \\ \left(\mathbf{R}_{t,\text{w}}^{(\pi(i))}\right)^{\top} \mathbf{R}_{t,\text{w}}^{(i)}, & \text{otherwise.} \end{cases} \tag{12}$$

(11) follows from orthogonality of $\mathbf{B}_{\text{rest}}^{(i)}$, and (12) is the standard parent-relative composition identity. Empirically, across 0.98 M 22-joint data frames, local rotation recovery $\phi(\cdot)$ is numerically exact up to float32 precision, with maximum matrix-space error $\max_t |\mathbf{R}_{t,\text{loc}} - \phi(\mathbf{R}_{t,\text{bone}})|_F$ = 1.8e–5; (mean 1.0e–5).

3.3 IK-GAT: Graph-attention inverse kinematics

Figure 3 summarizes the proposed IK-GAT architecture. We learn a mapping from root-space joint positions to rotation matrices representing bone-aligned world frames:

$$\tilde{\mathbf{J}}_t \mapsto \left\{\widehat{\mathbf{R}}_{t,\text{bone}}^{(i)}\right\}_{i=0}^{N-1}. \tag{13}$$

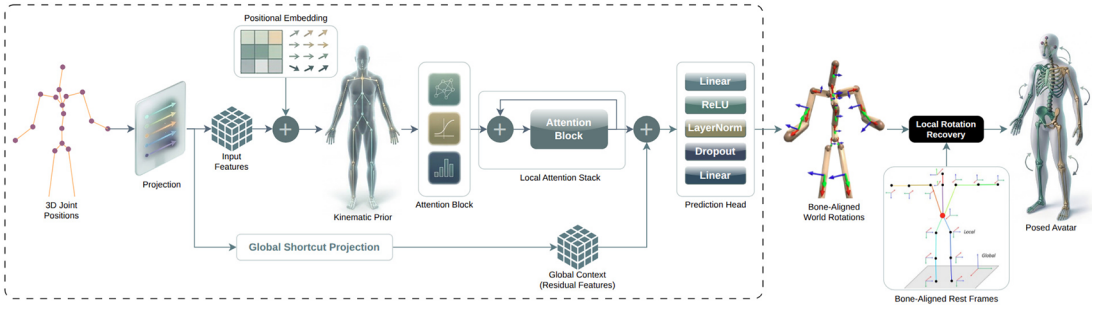

Figure 3. IK-GAT architecture. The model accepts 3D joint positions, projects them into a latent space with learnable positional embeddings, and applies graph attention structured by the skeletal parent-child hierarchy. Dual residual connections improve optimization, while the prediction head regresses bone-aligned world rotations using a continuous 6D representation.

3.3.1 Skeletal graph (kinematic prior)

The kinematic tree can also be denoted by a graph G = (V, E) with one node per joint and edges connecting each joint to its parent. We use the kinematic tree as an undirected message-passing graph by adding bidirectional edges $(i, \pi(i))$ and $(\pi(i), i)$ for each non-root joint. This imposes an inductive bias that information flows along anatomically meaningful connections.

3.3.2 Node features and embedding

Each joint is represented by a feature vector comprising its observed root-space position $\mathbf{x}^{(i)} = \tilde{\mathbf{j}}_t^{(i)} \in \mathbb{R}^3$, optionally concatenated with additional features. We then project the feature vector to a latent dimension F and add a learnable joint-index embedding:

$$\mathbf{h}_0^{(i)} = \mathbf{W}_{\text{in}}\mathbf{x}^{(i)} + \mathbf{e}^{(i)}. \tag{14}$$

This allows the network to specialize behavior per anatomical joint even under symmetric or ambiguous positional patterns.

3.3.3 Graph-attention blocks with distal refinement

We apply D stacked multi-head Graph Attention layers[34] on the kinematic graph. In the implementation, each GAT layer operates on the bidirectional parent-child graph and also includes self-loops, so each joint aggregates both its own feature and messages from its kinematic neighbors. For layer l,

$$\tilde{\mathbf{h}}_{l+1}^{(i)} = \text{LN}\left(\text{ELU}\left(\sum_{j \in \mathcal{N}^+(i)} \alpha_{ij}^{(l)} \mathbf{W}^{(l)} \mathbf{h}_l^{(j)}\right)\right), \tag{15}$$

where $\text{N}^+(i) = \text{N}(i) \cup \{i\}$ denotes the neighbor set augmented with the self-loop, $\alpha_{ij}^{(l)}$ are learned attention weights, and $\text{LN}(\cdot)$ is LayerNorm.

In addition, we use a *global* shortcut that bypasses the entire GAT stack. A learned linear projection of the raw per-joint input is added to the final pre-head features:

$$\mathbf{r}^{(i)} = \mathbf{W}_{\text{skip}}\mathbf{x}^{(i)}, \mathbf{z}^{(i)} = \mathbf{h}_D^{(i)} + \mathbf{r}^{(i)}. \tag{16}$$

This shortcut preserves direct access to the observed geometric input while the graph-attention layers model inter-joint dependencies.

When enabled, the model also uses a *local kinematic refinement* branch in the later half of the GAT stack rather than a standard hidden-state residual. Let $\text{C}(i)$ denote the children of joint i, and define the immediate kinematic neighborhood

$$\mathcal{K}(i) = (\{\pi(i)\} \text{if} \pi(i) \neq -1) \cup \mathcal{C}(i). \tag{17}$$

From previous-layer features, we compute the neighborhood mean

$$\overline{\mathbf{h}}_l^{(i)} = \frac{1}{|\mathcal{K}(i)|} \sum_{j \in \mathcal{K}(i)} \mathbf{h}_l^{(j)}, \tag{18}$$

and form a learned local correction

$$\Delta\mathbf{h}_l^{(i)} = \mathbf{W}_{\text{loc}}\left(\overline{\mathbf{h}}_l^{(i)} - \mathbf{h}_l^{(i)}\right). \tag{19}$$

This correction is applied only for layers in the later half of the stack and only on a distal-joint set D consisting of end effectors and their immediate parents:

$$\mathbf{h}_{l+1}^{(i)} = \tilde{\mathbf{h}}_{l+1}^{(i)} + 1[i \in \mathcal{D}]\Delta\mathbf{h}_l^{(i)}. \tag{20}$$

Thus, instead of a generic residual connection between consecutive hidden states, this is a structured, kinematic local refinement term that selectively adjusts distal chains such as wrist–elbow, ankle–foot, and neck–head, while proximal joints are handled primarily by graph message passing and the global shortcut.

3.3.4 Rotation head: 6D parameterization

Each joint feature is mapped to a 6D rotation representation $\widehat{\mathbf{r}}^{(i)} \in \mathbb{R}^6$, which is converted to a valid rotation matrix via Gram-Schmidt. Writing $\widehat{\mathbf{r}}^{(i)} = \left[\mathbf{a}_1^\top \;\; \mathbf{a}_2^\top\right]^\top$ with $\mathbf{a}_1, \mathbf{a}_2 \in \mathrm{P}^3$:

$$\widehat{\mathbf{x}}^{(i)} = \frac{\mathbf{a}_1}{\|\mathbf{a}_1\|_2}, \tag{21}$$

$$\tilde{\mathbf{y}}^{(i)} = \mathbf{a}_2 - \langle \mathbf{a}_2, \widehat{\mathbf{x}}^{(i)} \rangle \widehat{\mathbf{x}}^{(i)}, \widehat{\mathbf{y}}^{(i)} = \frac{\tilde{\mathbf{y}}^{(i)}}{\|\tilde{\mathbf{y}}^{(i)}\|_2}, \tag{22}$$

$$\widehat{\mathbf{z}}^{(i)} = \widehat{\mathbf{x}}^{(i)} \times \widehat{\mathbf{y}}^{(i)}, \tag{23}$$

and $\widehat{\mathbf{R}}_{t,\mathrm{bone}}^{(i)} = \left[\widehat{\mathbf{x}}^{(i)} \widehat{\mathbf{y}}^{(i)} \widehat{\mathbf{z}}^{(i)}\right] \in \mathrm{SO}(3)$.

3.4 Training objectives

Geodesic rotation loss. We supervise predicted rotation matrices using the geodesic distance on SO(3)[35]:

$$d_{\mathrm{geo}}(A, B) = \arccos\left(\frac{\mathrm{tr}\left(A^\top B\right) - 1}{2}\right), \tag{24}$$

where A, $B \in \mathrm{SO}(3)$, and

$$\mathcal{L}_{\mathrm{rot}} = \frac{1}{N} \sum_{i=0}^{N-1} d_{\mathrm{geo}}\left(\widehat{\mathbf{R}}_{t,\mathrm{bone}}^{(i)}, \mathbf{R}_{t,\mathrm{bone}}^{(i)}\right). \tag{25}$$

Since $-1 \leq \mathrm{tr}(U) \leq 3$ for any $U \in \mathrm{SO}(3)$, (25) is well-defined.

Forward-kinematics consistency. We optionally enforce that the predicted rotations reproduce the input root-space joints after analytic recovery of predicted world and local rotations (Equations 11–12) and forward kinematics (Equations 2–3). Let

$$\widehat{\mathbf{J}}_t^{\mathrm{rec}} = \left\{\widehat{\mathbf{j}}_t^{(i),\mathrm{rec}}\right\}_{i=0}^{N-1} = \mathrm{FK}\left(\left\{\widehat{\mathbf{R}}_{t,\mathrm{loc}}^{(i)}\right\}_{i=0}^{N-1}\right) \tag{26}$$

denote the root-space joints reconstructed from the predicted local rotations. We then define

$$\mathcal{L}_{\mathrm{fk}} = \frac{1}{N} \sum_{i=0}^{N-1} \|\widehat{\mathbf{j}}_t^{(i),\mathrm{rec}} - \widehat{\mathbf{j}}_t^{(i)}\|_2^2. \tag{27}$$

The total loss is

$$\mathcal{L} = \mathcal{L}_{\mathrm{rot}} + \alpha \mathcal{L}_{\mathrm{fk}}, \tag{28}$$

where α is a free hyperparameter controlling the influence of the FK consistency regularization.

3.5 Inference and deployment

At inference time, the network predicts $\widehat{\mathbf{R}}_{t,\mathrm{bone}}^{(i)}$ from the root-space input $\tilde{\mathbf{J}}_t$ in a single pass. We recover predicted world rotation matrices via (11) and then predicted parent-relative local rotation matrices via (12). Because the network operates in root space, the global root translation $\mathbf{t}$ is not predicted by the model and is added separately when world-space placement is required.

Given a global root translation $\mathbf{t}$, world-space joint positions can be obtained as

$$\mathbf{J}_t = \left\{ \widehat{\mathbf{j}}_t^{(i)} + \mathbf{t} \right\}_{i=0}^{N-1}. \tag{29}$$

Since the IK mapping depends only on relative joint configuration, we operate entirely in root space during learning and inference.

The only rig-specific inputs required are the kinematic tree and the rest-frame rotation matrices $\left\{ \mathbf{B}_{\text{rest}}^{(i)} \right\}$, computed once per rig. The resulting local rotations can be applied directly to a rigged avatar, or converted to axis–angle parameters for parametric body models such as SMPL.[19] In that case, the recovered root rotation $\widehat{\mathbf{R}}_{t,\text{loc}}^{(0)} = \widehat{\mathbf{R}}_{t,\text{w}}^{(0)}$ corresponds to the global orientation parameter, the non-root local rotations $\left\{ \widehat{\mathbf{R}}_{t,\text{loc}}^{(i)} \right\}_{i=0}^{N-1}$ correspond to the other pose parameters, and the root translation $\mathbf{t}$ corresponds to the global translation parameter. Shape coefficients $\boldsymbol{\beta}$ are not estimated by IK-GAT and must come from an external source (e.g., a fixed mean shape, a subject-specific estimate, or a downstream fitting step).

4. Experiments

We evaluate IK-GAT as an amortized inverse-kinematics module that sits between upstream 3D joint tracking and downstream avatar animation. The experiments address four questions: whether single-pass inference can match or exceed iterative IK on a standardized full-body benchmark; whether the proposed bone-aligned world representation simplifies learning relative to direct local-rotation regression; whether anatomically structured graph attention is more effective than generic dense interaction; and whether the resulting model transfers to a production rig under noisy tracking. We therefore use two complementary benchmarks. The SMPL benchmark isolates pose recovery under standardized kinematics and provides the main quantitative comparison. The UE5 benchmark tests whether the analytically recovered local rotations remain directly useful on a production rig. Unless otherwise stated, all models operate causally on single frames of root-space joints, without future context or iterative refinement.

4.1 Evaluation metrics

Our primary metric is **mean per-joint angular error (MPJAE)** in parent-relative local rotation space. This aligns the evaluation with the final animation-ready rotation representation recovered at inference time. Given predicted bone-aligned world rotations $\widehat{\mathbf{R}}_{t,\text{bone}}^{(i)}$ and ground-truth bone-aligned world rotations $\mathbf{R}_{t,\text{bone}}^{(i)}$, we first recover the corresponding local rotations $\widehat{\mathbf{R}}_{t,\text{loc}}^{(i)}$ and $\mathbf{R}_{t,\text{loc}}^{(i)}$ using Equations 11–12. We then define

$$\text{MPJAE} = \frac{180}{\pi N} \sum_{i=1}^{N-1} \arccos\left(\frac{\text{tr}\left(\left(\widehat{\mathbf{R}}_{t,\text{loc}}^{(i)} \right)^{\top} \mathbf{R}_{t,\text{loc}}^{(i)} \right) - 1}{2} \right), \tag{30}$$

reported in degrees, and averaged over all evaluation samples.

We also report **swing** and **twist** errors as secondary rotation diagnostics derived directly from the predicted and ground-truth bone-aligned world frames. Swing error is measured as the angular distance between the predicted and ground-truth first frame axes, $\widehat{\mathbf{x}}^{(i)}$ and $\mathbf{x}^{(i)}$, while twist error is measured as the angular distance between the corresponding second frame axes, $\widehat{\mathbf{y}}^{(i)}$ and $\mathbf{y}^{(i)}$. These diagnostics are useful because they reveal whether an improvement comes primarily from recovering bone direction or from resolving the residual in-plane ambiguity around the bone axis.

For the SMPL benchmark, we additionally report **MPJPE** and **P-MPJPE** on joints, and **MPVE** and **P-MPVE** on mesh vertices when a body-model reconstruction is available. Let predicted and ground-truth joints be $\widehat{\mathbf{J}}_t = \left\{ \widehat{\mathbf{j}}_t^{(i)} \right\}_{i=0}^{N-1}$ and $\mathbf{J}_t = \left\{ \mathbf{j}_t^{(i)} \right\}_{i=0}^{N-1}$, with $\widehat{\mathbf{j}}_t^{(i)}, \mathbf{j}_t^{(i)} \in \mathbb{R}^3$. We define

$$\text{MPJPE} = 1000 \cdot \frac{1}{N} \sum_{i=0}^{N-1} \| \widehat{\mathbf{j}}_t^{(i)} - \mathbf{j}_t^{(i)} \|_2, \tag{31}$$

reported in millimeters. P-MPJPE first aligns the predicted joints to the ground truth using a Procrustes similarity transform.[36] MPVE and P-MPVE are defined analogously on mesh vertices.

These geometry metrics are *secondary* in our setting because they depend not only on pose quality but also on how body shape is handled. MPJAE is therefore the main metric for comparing IK performance, while joint and mesh errors are used to assess downstream compatibility with SMPL-style reconstruction pipelines.

4.2 SMPL animation benchmark

Protocol. We use AMASS[37] as the primary benchmark. For each frame we recover 22 SMPL body joints in root space and compute the corresponding bone-aligned world rotations using Equation 10, yielding paired supervision targets $(\tilde{\mathbf{J}}_t, \{\mathbf{R}^{(i)}_{t,\text{bone}}\}_{i=0}^{N-1})$ with maximum canonical axis-angle invertibility error 7.96e–6; (mean 1.84e–6). We train on the AMASS CMU subset, validate on HDM05, HumanEva, MoSh, SFU subsets, and report final numbers on SSM and Transitions subsets. The dataset comprises 1 M train frames, 0.33 M validation frames, and 0.03 M test frames, respectively. The model predicts pose only. Unless noted otherwise, body-model metrics for feed-forward methods are computed using ground-truth shape so that the benchmark isolates pose recovery.

Baselines. We establish a lower bound by evaluating metrics with all parameters fixed to zero, and compare against three optimization baselines (LBFGS,[26] VPoser,[26] and SMPLify3D[38]) and five feed-forward baselines (MLP, QuaterGCN,[39] QuaterGCN-6D, STIK, and a dense Transformer). LBFGS is identical to VPoser without the human body prior. All feed-forward models consume the same 22 root-space joints and are trained to predict the same rotation targets. Further details about the neural baselines are provided in the supplementary materials. We omit numerical comparisons with HybrIK,[10] MANIKIN,[15] and HierarIK[8] as they rely on visual cues or 6-DoF orientation priors absent in our pure 3D positional setting, as well as the lack of reproducible code for the latter two for fair evaluation.

Implementation details. All models are trained on the same data splits with the same preprocessing and early-stopping criterion, so performance differences can be attributed to the model design rather than mismatched training procedures. Our default configuration uses hidden dimension $F = 256$, $D = 4$ graph-attention blocks, and 8 attention heads. We train with AdamW, batch size 512, initial learning rate 0.001, weight decay 0.01 (bias, norm and embedding has no decay), dropout 0.1, and FK-consistency weight $\alpha = 0.1$. Training runs for up to 10 epochs with early stopping if validation MPJAE does not improve for 3 consecutive epochs. Parameter counts and multiply-adds are reported for batch size 1.

4.2.1 Main results

Table 1 shows that the main advantage of IK-GAT is not only lower error, but a better alignment between the learned representation and the actual IK objective. The strongest generic baseline is the dense Transformer, which confirms that full-body rotation recovery requires global context and cannot be solved by independent joint-wise regression. However, once global interaction is already available, unrestricted attention is still suboptimal: IK-GAT improves test MPJAE from 9.06 to 7.43 and test MPVE from 33.08 to 24.47 while using only 0.37 M parameters instead of 3.17 M. This indicates that the remaining gap is not due to insufficient model capacity, but due to the absence of the two priors encoded by IK-GAT: a canonical bone-aligned rotation space and anatomically constrained message passing.

Among the iterative methods, VPoser is strongest on MPJAE, while SMPLify3D is stronger on geometry. This split reflects an objective mismatch: optimization can exploit pose-shape trade-offs to improve joint and mesh reconstruction without necessarily recovering better local rotations. Since our goal is animation-ready IK rather than generic body-model fitting, MPJAE is the more informative metric here. Under that metric, IK-GAT outperforms all iterative baselines despite not performing any per-frame optimization. In other words, amortization is not only reducing runtime; it is recovering a better pose variable for the task we actually care about. Figure 4 supports this observation qualitatively. Although IK-GAT takes only 3D joint positions as input, the recovered poses align closely with the ground-truth mesh. When evaluated using ground-truth shape parameters (top row), the remaining error is mainly due to small pose inaccuracies, whereas with fixed shape (bottom rows), visible shape mismatch appears but articulation remains consistent. This further indicates that IK-GAT primarily learns a stable inverse-kinematics mapping rather than relying on implicit shape cues.

A third pattern is the unusually large gap between weak and strong neural baselines. MLP fails catastrophically, and even QuaterGCN-6D remains far behind the Transformer and IK-GAT. This suggests that the hard part of dense IK is not simply rotation parameterization, but structured disambiguation: the network must decide which joints should influence which others, and in what coordinate system the prediction problem is easiest to solve. IK-GAT performs best because it resolves both issues simultaneously. The graph restricts information flow to plausible anatomical pathways, while bone-world supervision removes rig-specific basis ambiguity from the target space.

Table 1. AMASS SMPL benchmark. Lower is better. IK-GAT achieves the best overall performance. Compared with the strongest feed-forward baseline (Transformer), it reduces test MPJAE from 9.06 to 7.43 (−18.0%) while using 8.6× fewer parameters (0.37 M vs. 3.17 M). Compared with the strongest iterative baseline on MPJAE (VPoser), it further reduces test MPJAE from 9.24 to 7.43 despite requiring only a single forward pass instead of 200 optimization steps.

	Validation Set					Test Set					Runtime	
Method	**MPJAE**	**MPJPE**	**P-MPJPE**	**MPVE**	**P-MPVE**	**MPJAE**	**MPJPE**	**P-MPJPE**	**MPVE**	**P-MPVE**	**Params(M)**	**MAdds(M)**
Lower Bound	22.95	691.03	160.54	849.55	211.32	23.88	650.99	136.47	835.76	173.71	-	-
LBFGS	16.92	45.34	17.38	57.15	28.59	14.93	40.42	**11.30**	51.75	30.79	-	-
VPoser	10.25	50.05	22.70	63.04	33.30	9.24	44.13	23.07	59.57	38.35	-	-
SMPLify3D	10.61	19.58	**9.56**	34.72	24.99	11.86	34.61	14.03	48.91	31.99	-	-
MLP‡	45.82	227.23	136.19	306.74	151.72	39.32	211.70	117.86	299.64	130.95	2.11	2.23
QuaterGCN†	16.64	56.94	41.20	73.26	44.44	14.72	48.18	36.37	57.44	37.76	0.13	2.70
QuaterGCN-6D†	15.35	44.26	34.16	56.28	37.28	14.31	37.94	31.78	44.19	30.51	0.13	2.71
STIK‡	13.83	50.41	37.95	57.73	37.93	12.57	43.05	34.51	44.93	31.90	1.32	174.87
Transformer‡	9.48	22.86	17.40	37.61	27.70	9.06	22.54	16.53	33.08	24.55	3.17	2.23
IK-GAT (ours)	**7.98**	**18.19**	14.21	**27.90**	**21.73**	**7.43**	**18.60**	12.80	**24.47**	**18.08**	0.37	6.10

Open Research Europe 2026, 6:314 Last updated: 01 OCT 2026

Per-joint comparisons. Figure 5 compares IK-GAT with the strongest non-graph baseline at the joint level. The largest improvements appear on distal and twist-sensitive joints, where local evidence is weakest and ambiguity must be resolved from broader kinematic context. Gains are smaller on pelvis and hips because their orientations are already strongly constrained by adjacent joints. Wrists remain difficult for all methods as expected: leaf nodes provide almost no positional evidence for twist. This distribution mirrors the design of IK-GAT itself. The bidirectional graph propagates context along the chain, while the distal local refinement explicitly targets the joints where independent regression fails most.

Iteration-budget tradeoff. Figure 6 isolates the cost of iterative fitting. The optimization curves only approach IK-GAT after tens to hundreds of iterations, and their gains beyond that point are larger on joint and mesh metrics than on local-space MPJAE. This divergence is informative: once shape and global body configuration can absorb residual error, geometry can continue improving even when local rotations do not. In contrast, IK-GAT directly targets the animation variable of interest.

Runtime. Single-frame inference reaches 693 FPS on CPU and 448 FPS on GPU (RTX 4060), while GPU provides up to 3.7× higher throughput under batching. Further information is provided in the supplementary materials.

4.2.2 Ablation study

We ablate the output representation, the auxiliary FK-consistency loss, the graph structure, the shortcut mechanisms, and the model capacity. All ablations use the same training protocol as the main model unless stated otherwise.

Effect of bone-aligned world rotation space. Training directly in parent-relative local space forces the network to learn both pose and rig-specific basis conventions at the same time (Table 2). Bone-aligned world space removes this nuisance variable by canonicalizing the local frame once, outside the network. The resulting gains are not limited to MPJAE. They are even larger on MPJPE and MPVE after local recovery, which indicates that the representation improves global kinematic consistency rather than merely making angular regression numerically easier.

Effect of graph structure and attention. Figure 7a shows that graph structure is a first-order design choice. Fully connected attention weakens the anatomical bias by encouraging interactions between unrelated joints, while a unidirectional graph prevents information from propagating back along the chain. Replacing attention with standard graph convolution also hurts performance, as neighbors are then treated too uniformly. Best result comes from the combination of bidirectional kinematic graph and learned attention weights.

Effect of IK-GAT components. Table 3 shows that positional embeddings are indispensable: without joint identity, the network struggles to distinguish anatomically distinct but geometrically similar configurations, and MPJAE degrades

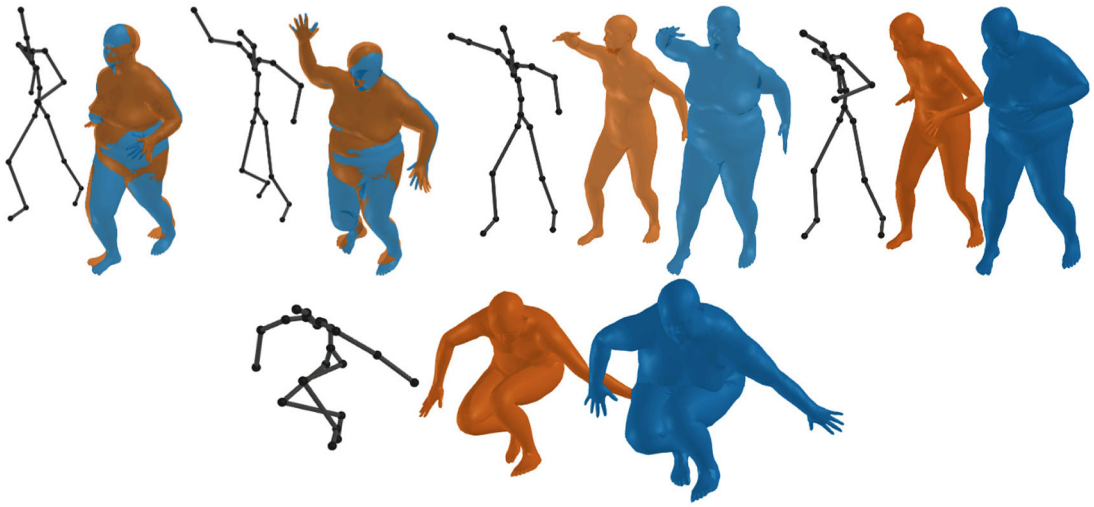

Figure 4. SMPL qualitative results. Input 3D joints (black) and reconstructed meshes (brown) are compared with the ground-truth mesh (blue). The top row uses ground-truth shape parameters, while the remaining rows use a fixed body shape across samples, illustrating that articulation remains accurate even under shape mismatch.

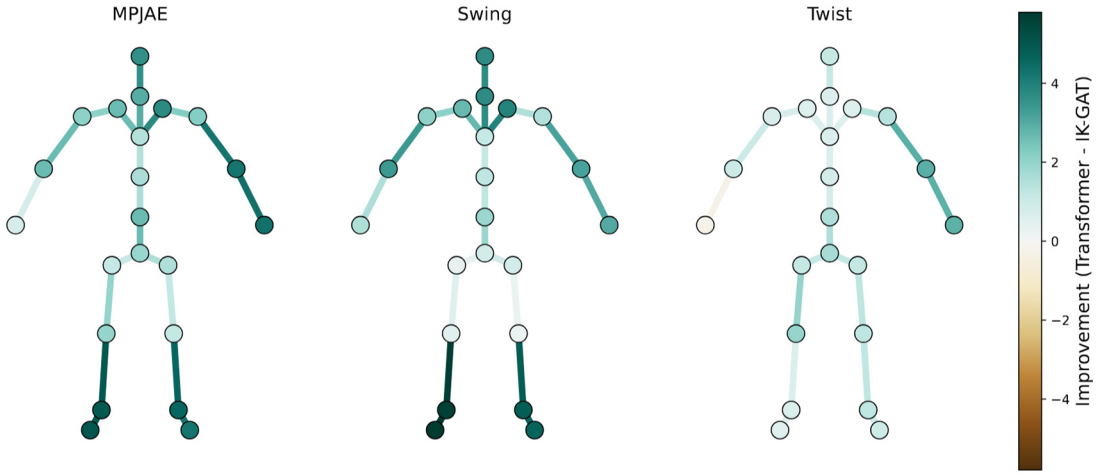

Figure 5. Per-joint improvement over the strongest dense baseline. Improvements concentrate on distal and twist-sensitive joints, including the ankles, feet, elbows, neck, and head, showing where kinematic message passing and distal refinement provide the greatest benefit. Forearm twist remains challenging because positional evidence for axial rotation is inherently limited without tracking the hand joints.

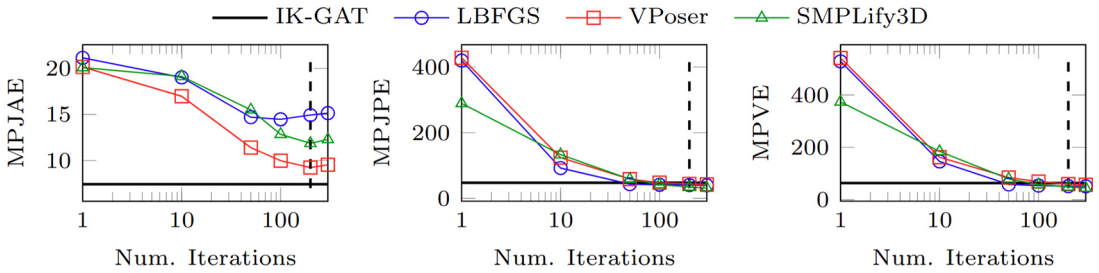

Figure 6. Iteration budget versus accuracy on AMASS. Iterative optimization requires tens to hundreds of iterations to approach the accuracy achieved by a single forward pass of IK-GAT, highlighting the benefit of amortized inference.

Table 2. Rotation-space ablation. Training in bone-aligned world space consistently outperforms direct local-rotation regression, particularly on geometry-aware metrics, indicating that the proposed representation simplifies learning without sacrificing local rotation accuracy.

Rotation Space	MPJAE	MPJPE	P-MPJPE	MPVE	P-MPVE
Parent-Relative	7.93	31.68	23.49	47.31	29.29
Bone-World	7.48	18.61	12.80	24.48	18.08

sharply. Once embeddings are present, both shortcut mechanisms offer smaller but complementary gains. The global shortcut preserves direct access to the observed geometry, while the local refinement improves the parts of the body where local ambiguity accumulates most strongly.

Effect of FK-consistency loss. Figure 7b studies the influence of the FK-consistency weight α. A small auxiliary term is beneficial because it couples the predicted rotations back to the observed joint geometry and discourages locally plausible but globally inconsistent solutions. However, once α becomes too large, training starts to chase joint reconstruction too aggressively and the trade-off between rotational and geometric metrics becomes unstable. We therefore use $\alpha = 0.1$ as the most reliable compromise.

Effect of model size. Figure 8 shows a clear diminishing-return regime. Scaling from Tiny to Base (default) yields the expected improvement, but larger models do not improve the primary MPJAE metric and even slightly worsen MPVE.

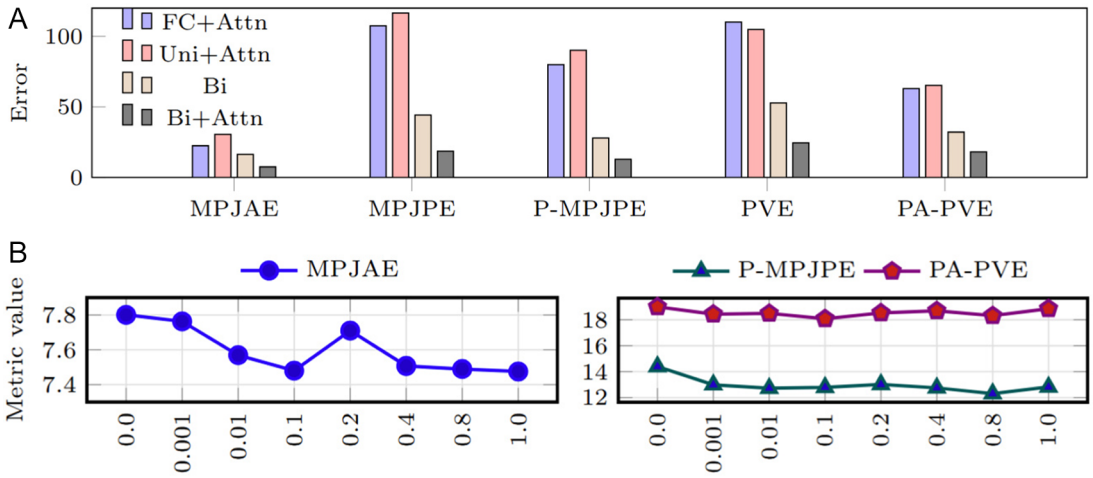

Figure 7. Ablation study. Effects of graph connectivity, attention, and FK-consistency regularization on inverse-kinematics performance. (a) Ablation of graph structure and attention. Bidirectional kinematic message passing with attention is critical. Fully connected graphs dilute the anatomical inductive bias, one-way graphs block corrective feedback along the chain, and non-attentive graph propagation cannot adapt neighbor importance to the current pose. (b) Effect of FK-consistency weight α. A small FK regularization improves both local-rotation accuracy and downstream geometry by tying the predicted rotations back to the observed joints. Larger weights over-constrain training and make the optimization trade-off increasingly unstable, so we use $\alpha = 0.1$ throughout.

Table 3. Component ablation under the bidirectional graph. Positional embeddings (**PE**) are essential as they disambiguate anatomically distinct joints under similar local geometry. The global shortcut (**GS**) and distal local refinement (**LR**) provide smaller but complementary gains, with the complete model achieving the best overall performance.

PE	GS	LR	MPJAE	MPJPE	P-MPJPE	MPVE	P-MPVE
✓			7.58	19.02	13.65	**24.29**	18.40
	✓		15.68	44.94	33.09	57.83	38.34
		✓	16.31	45.09	34.24	56.80	38.24
	✓	✓	17.26	45.00	33.13	54.78	37.08
✓		✓	7.51	**18.01**	13.58	24.72	19.31
✓	✓		7.64	19.04	**12.75**	25.87	18.65
✓	✓	✓	**7.48**	18.61	12.80	24.48	**18.08**

Once the representation and the kinematic prior are fixed, additional capacity does not resolve the underlying ambiguity more effectively.

4.3 Game-rig animation benchmark

The second benchmark addresses a different question: not whether the recovered pose best explains a parametric body model, but whether the analytically recovered local rotations are directly usable on a production rig in a realistic deployment setting. We therefore evaluate IK-GAT on the UE5 mannequin, with no body model in the loop and no retargeting at inference time.

Benchmark setup and motivation. Our target application is remote physiotherapy,[40] where a tracked person performs full-body exercises and the motion is streamed to an avatar in Unreal Engine. This setting is challenging for two reasons. First, many target motions, especially low-posture rehabilitation exercises such as planks and bridges, are underrepresented in public motion datasets. Second, public Unreal-oriented datasets[41] often target older skeleton conventions, so retargeting would introduce an additional source of error that is orthogonal to the inverse-kinematics problem studied here. We therefore construct a benchmark directly in the UE5 mannequin convention and evaluate whether IK-GAT transfers beyond the standardized SMPL setting. In end-to-end deployment illustrated in Figure 9, a multi-view

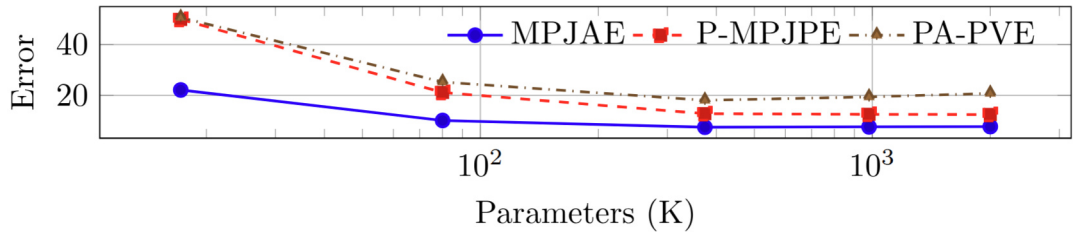

Figure 8. Model scaling. Performance improves rapidly from Tiny to Base before largely saturating, indicating that most gains arise from the proposed representation and anatomical inductive bias rather than increased model capacity.



Figure 9. Real-time deployment pipeline. Markerless multi-view tracking first estimates dense 3D joint positions, which IK-GAT maps to bone-aligned world rotations before analytically recovering UE5-Mannequin local joint rotations for direct rendering in Unreal Engine. The benchmark therefore evaluates the complete animation-facing inverse-kinematics pipeline rather than body-model reconstruction.

markerless tracker[42] provides dense 3D joints, IK-GAT converts these joints into animation-ready local rotations, and the recovered rotations are streamed to Unreal Engine for real-time rendering.

UE-IK dataset construction. To train and evaluate the method in this rig convention, we construct the *UE-IK* dataset comprising paired root-space joint positions and local joint rotations in the UE5 mannequin skeleton. The data, summarized in Table 4, combines two complementary sources. First, we collect a large set of high-quality animation clips from Unreal Engine's marketplace and export per-frame joint positions and local bone rotations through the Unreal Python API. These clips provide broad motion diversity and clean supervision in the exact target skeleton convention. Second, to better reflect the target rehabilitation scenario, we record real people using an Xsens IMU-based motion-capture system,[43] stream the motion to the UE5 mannequin through LiveLink, and export the resulting supervision data in the same format. This second source introduces motions that are both closer to deployment and less well represented in generic public datasets. The real-world recording protocol, illustrated in Figure 10, covers twelve subjects. Each subject performs three types of motion: full-range-of-motion articulation for major body parts, repeated target rehabilitation exercises, and short free-form movement sequences.

Training protocol and transfer. All models are initialized from their corresponding AMASS-pretrained checkpoints and then adapted to the UE-IK benchmark. This transfer step is important because UE-IK is much smaller than the SMPL benchmark and differs not only in motion distribution but also in skeleton topology and joint count. To make transfer effective across skeletons, we remap learned positional embeddings from the source skeleton to the corresponding destination joints, initialize unmatched joints with zeros, and partially reuse pretrained weights for layers whose structure depends on the number of joints. This preserves a large fraction of the learned anatomical prior and substantially improves convergence under limited target data.

Table 4. UE-IK dataset statistics. The training split combines marketplace animations with real-world IMU-driven UE5 sequences, while the validation and test splits emphasize rig-specific articulation and deployment-relevant motions. This evaluates transfer from predominantly synthetic animation to real recorded movements within the same target skeleton convention.

Split	Sequences/ Frames	Notes
Train	491/305 k	80% of our IMU-based real-world data (59 sequences, 158 k frames), combined with marketplace animations covering dancing, gym workouts, yoga, locomotion, walking, and other everyday motions.
Val	40/1 k	Built-in UE5 male (Manny) and female (Quinn) avatars, each articulating individual bones through their full range of motion.
Test	17/49 k	20% of the real-world IMU-based data, including free movements, target rehabilitation workouts, and range-of-motion sequences.

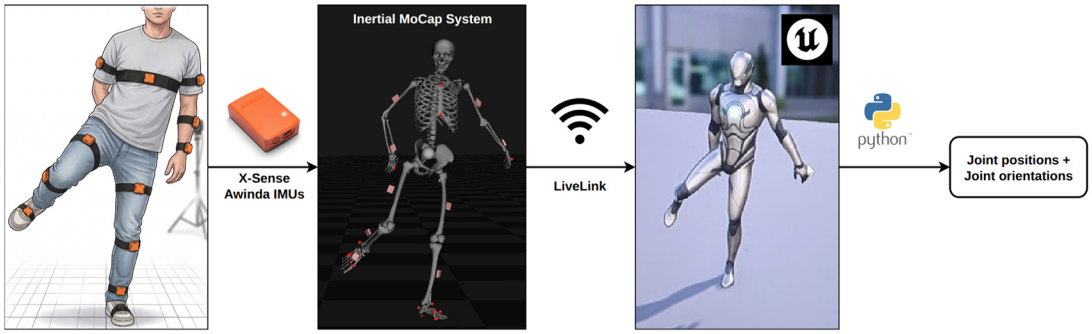

Figure 10. UE-IK acquisition process. Real motion is captured with Xsens IMUs, streamed to UE, and exported as paired joint positions and rotations. Marketplace animations are exported through the same pipeline, giving unified training data for the benchmark.

Main results. Table 5 shows that the ranking on the UE5 benchmark mirrors the main SMPL study, but with a sharper separation between models that explicitly encode structure and those that do not. The MLP baseline fails by a wide margin, which reinforces the central claim of the paper: dense position-to-dense orientation recovery is not a simple regression task. Even when every joint is observed, the output must satisfy a coupled kinematic structure and resolve twist ambiguity in a rig-consistent way. Purely token-wise regression is therefore insufficient. Introducing more structure improves performance progressively. QuaterGCN-6D is substantially stronger than the MLP, indicating that graph-based propagation is already beneficial. STIK improves further by adding temporal and parent-conditioned modeling, which is consistent with the intuition that even weak kinematic cues are useful when data is limited. The Transformer is the strongest baseline, showing that global context is important on this benchmark. However, IK-GAT still achieves the best result despite using a much smaller parameter budget than the Transformer. This is the key result of the UE benchmark: once the target is a real production rig, anatomically constrained message passing and exact local-rotation recovery are more valuable than generic dense attention alone. The gap is also informative when decomposed into swing and twist. The improvement in swing indicates that the graph prior helps reconstruct globally coherent limb directions under cross-skeleton transfer. The remaining gain in twist, although smaller in absolute terms, is the more meaningful one: twist is exactly the part of the problem that is weakly constrained by positions alone, especially under distribution shift.

Figure 11a complements the quantitative results in Table 5 by demonstrating that the recovered local rotations can be used directly inside Unreal Engine, preserving kinematic chain coherence and plausible articulation without any additional post-processing. Figure 11b further evaluates the method on a more challenging rehabilitation-style motion. The top row shows the tracked pose and reconstructed skeleton provided as input to IK-GAT, while bottom row shows the resulting UE5 avatar animation driven entirely by the analytically recovered local joint rotations. Consistent with the quantitative results, the transferred motion remains coherent and directly usable for animation without retargeting or body-model fitting, even for low-posture poses where recovering joint rotations from positions is inherently more ambiguous.

Table 5. UE5 mannequin benchmark. Lower is better. All models are initialized from their AMASS-pretrained checkpoints and fine-tuned on UE-IK. IK-GAT achieves the best local-space MPJAE together with the lowest swing and twist errors, demonstrating effective transfer of the proposed representation and kinematic message passing to a production animation rig.

Method	MPJAE	Swing	Twist	Params(M)
MLP	75.08	38.27	57.79	2.11
QuaterGCN-6D	30.33	22.01	19.29	0.13
STIK	20.64	12.15	14.25	1.32
Transformer	15.61	7.39	11.93	3.17
IK-GAT (ours)	**12.83**	**5.20**	**10.42**	0.37

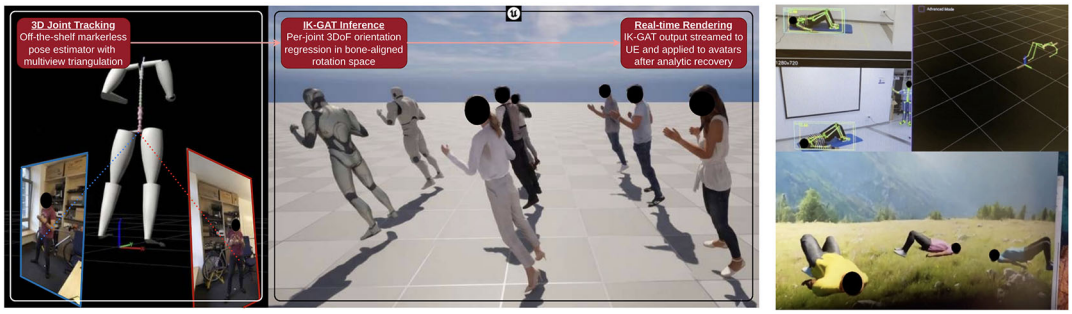

Figure 11. Qualitative UE5 results. IK-GAT predictions are streamed directly to Unreal Engine avatars without optimization or manual post-processing. Left: examples across multiple avatars demonstrating consistent animation quality. Right: a rehabilitation-style exercise showing the tracked pose, reconstructed skeleton, and resulting avatar animation, illustrating robustness on uncommon low-posture motions that are particularly sensitive to articulation errors.

5. Conclusion

We presented IK-GAT, a lightweight inverse-kinematics model that reconstructs full-body joint orientations from 3D joint positions in a single forward pass. The key idea is to separate the learning problem from the kinematic conversion: the network predicts bone-aligned world rotations, while the transformation to standard parent-relative joint rotations is recovered analytically from the known rest pose and skeletal hierarchy. This decomposition reduces the complexity of the learned component and yields a representation that is stable for regression while remaining exactly compatible with conventional animation pipelines. Across both standardized and deployment-oriented benchmarks, the results support the same conclusion: dense-joint IK benefits most from the combination of a canonical rotation space, an explicit anatomical prior, and exact recovery to local rotations. The gains are especially pronounced on distal and twist-sensitive joints, where generic feed-forward regression and per-sample optimization are weakest. Because the method does not require iterative fitting or body-model optimization at inference time, it is well suited for online avatar animation, telepresence, live virtual production, and interactive character control. The current formulation still inherits three limitations of position-only IK: twist remains intrinsically ambiguous in rare or out-of-distribution poses, the analytic recovery assumes a known rest pose and skeletal topology, and the default model is single-frame rather than explicitly temporal. These limitations point to the most promising next steps: short-horizon temporal context, additional sensing cues such as IMUs or surface orientation, and topology-robust transfer across heterogeneous rigs.

Software availability

The source code of the proposed IK-GAT model is published via the data repository. This also includes an exact definition of the skeleton rig used for the animation in Unreal Engine.

- **skeletons.py**: UE rig definition
- **graph_ik.py**: PyTorch implementation of IK-GAT
- **coarse_ik_layer.py**: Coarse rotations from joint positions used to initialize IK-GAT for improved performance

- **causal_temporal_conv.py**: Experimental temporal module for IK-GAT (not used in main experiments)

The code is available under CC-BY 4.0.

Ethics and consent

The UE-IK dataset was collected on 31 July and 1 August 2025 from ten adult volunteers within the authors' research group. Participants wore a non-invasive inertial motion-capture system while performing predefined physical exercises. Before participation, all volunteers received information about the purpose of the study, the recording procedure, the intended use and retention of the collected data, and their rights as participants. Written informed consent was obtained from every participant before data collection.

Participation was voluntary, and participants could decline individual aspects of the study. The consent process covered video recording, wearable IMU sensor placement, long-term internal data retention, and optional public release of recordings only after facial anonymization. Participants could request a same-gender researcher for sensor placement where available. Participants who did not consent to longer-term retention could request deletion of their recordings after six months.

The released UE-IK dataset contains only motion-capture data and derived skeletal motion. Names, direct personal identifiers, images, and videos are not included in the public release. A copy of the participant information sheet and consent form is provided in the supplementary material available through the data repository.

At the time of data collection, the investigators considered the study to present no more than minimal risk because it involved competent adult volunteers, non-invasive wearable sensors, predefined physical exercises, and no medical intervention or deception. No prospective ethics-committee review was obtained.

The DFKI Ethics Committee subsequently confirmed in writing that, when the study was conducted, submission of research projects for ethical review was voluntary and was not a mandatory institutional requirement. Accordingly, prospective submission of the UE-IK study and receipt of ethics approval were not required under the institutional requirements in effect at that time. The Committee further confirmed that the applicable DFKI process did not require review by an external ethics committee where no mandatory institutional ethics review requirement applied.

Data availability

Underlying data

We use the publicly available AMASS dataset for core experiments in this work. The game rig experiments used the UE-IK dataset created in this work, and is available on Zenodo under CC-BY 4.0 license. The repository includes the following files:

Repository: K-GAT Model for Neural Inverse Kinematics and Unreal Engine IK (UE-IK) Dataset on Zenodo.[44]

Link: https://doi.org/10.5281/zenodo.21495883

- **train**: Training set
  - UE-IK_BlacksmithCarpenter.npz
  - UE-IK_DFKI-FreeMovements_train.npz
  - UE-IK_DFKI-HealthWorkouts_train.npz
  - UE-IK_DFKI-ROM_train.npz
  - UE-IK_DanceMoCap.npz
  - UE-IK_Free.npz
  - UE-IK_GymPack.npz

  - UE-IK_Locomotion.npz
  - UE-IK_ModernDance.npz
  - UE-IK_WalksMoCapVol2.npz
  - UE-IK_WorkAnimations.npz
  - UE-IK_WorkoutAnimations.npz
  - UE-IK_YogaAnimations.npz
- **val**: Validation set
  - UE-IK_Manny.npz
  - UE-IK_Quinn.npz
- **test**: Test set
  - UE-IK_DFKI-FreeMovements_test.npz
  - UE-IK_DFKI-HealthWorkouts_test.npz
  - UE-IK_DFKI-ROM_test.npz
- **supplementary.pdf**: Additional results and supplementary materials.

References

1. Alberto M: *Understanding motion capture for computer animation and video games.* Morgan kaufmann; 2000.
2. Weidong G, Gino Y: **Reuse of motion capture data in animation: A review.** *International Conference on Computational Science and Its Applications.* Springer; 2003; 620–629.
Publisher Full Text
3. Michael G: **Retargetting motion to new characters.** *Proceedings of the 25th annual conference on Computer graphics and interactive techniques.* 1998; 33–42.
4. Ladislav K, Steven C, Jiří Ž, *et al.*: **Skinning with dual quaternions.** *Proceedings of the 2007 symposium on Interactive 3D graphics and games.* 2007: 39–46.
5. Nadine Abu Rumman and Marco Fratarcangeli: **Skinning techniques for articulated deformable characters.** *Computer Graphics Theory and Applications.* 2016.
6. Enrico P, Sung-Kyu Y, Andreas A, *et al.*: **Sparseposer: Real-time full-body motion reconstruction from sparse data.** ***ACM Transactions on Graphics.*** *To appear/issue details vary by venue listing.* 2023; **43**: 1–14.
Publisher Full Text
7. Tom Van W, Seunghwan L, Antoine F, *et al.*: **Diffusionposer: Real-time human motion reconstruction from arbitrary sparse sensors using autoregressive diffusion.** *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2024; 2513–2523.
8. Yi X, Zhou Y, Feng X: **Hierarik: Hierarchical inverse kinematics solver for human body and hand pose estimation.** *CAAI International Conference on Artificial Intelligence (CICAI), Lecture Notes in Computer Science.* Springer; 2021. online date listed as 2022 in publisher metadata.
9. Gene SL, Lin A, Schiller M, *et al.*: **Enhanced dual quaternion skinning for production use.** *SIGGRAPH Talks.* 2013; 9–1.
10. Li J, Xu C, Chen Z, *et al.*: **Hybrik: A hybrid analytical-neural inverse kinematics solution for 3D human pose and shape estimation.** *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2021.
11. Gildea K, Mercadal-Baudart C, Blythman R, *et al.*: **Kinepose: A temporally optimized inverse kinematics technique for 6dof human pose estimation with biomechanical constraints.** *24th Irish Machine Vision and Image Processing Conference.* Irish Pattern Recognition & Classification Society; 2022; 105–112.
12. Buss SR: *Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. Technical report.* San Diego: University of California; April 2004. Technical report; made available online by the author.
13. Wang L-CT, Chen CC: **A combined optimization method for solving the inverse kinematics problem of mechanical manipulators.** *IEEE Trans. Robot. Autom.* 1991; **7**(4): 489–499.
Publisher Full Text
14. Aristidou A, Lasenby J: **Fabrik: A fast, iterative solver for the inverse kinematics problem.** *Graph. Model.* 2011; **73**(5): 243–260.
Publisher Full Text
15. Jiang Z, Chen Z, Zhou Y, *et al.*: **MANIKIN: Neural inverse kinematics from sparse signals.** *European Conference on Computer Vision (ECCV).* 2024. Preprint/Camera-ready may differ.
16. Li J, Bian S, Liu Q, *et al.*: **Niki: Neural inverse kinematics with invertible neural networks for 3D human pose and shape estimation.** *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2023.
17. Zhou Y, Barnes C, Lu J, *et al.*: **On the continuity of rotation representations in neural networks.** *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 2019.
18. Bensadoun R, Gur S, Blau N, *et al.*: **Neural inverse kinematics.** *Proceedings of the 39th International Conference on Machine Learning*

(ICML), volume 162 of Proceedings of Machine Learning Research. 2022; 1787–1797.

19. Loper M, Mahmood N, Romero J, *et al.*: **Smpl: A skinned multi-person linear model.** *ACM Trans. Graph.* 2015; **34**(6): 1–16.
Publisher Full Text

20. Khan MSU, Wang C-Y, Prokosch T, *et al.*: **Amortized inverse kinematics via graph attention for real-time human avatar animation.** 2026. arXiv preprint arXiv:2604.16629.

21. Donald Lee Pieper: *The Kinematics of Manipulators under Computer Control.* Stanford University: 1968. PhD thesis.

22. Delp SL, Anderson FC, Arnold AS, *et al.*: **Opensim: Open-source software to create and analyze dynamic simulations of movement.** *IEEE Trans. Biomed. Eng.* 2007; **54**(11): 1940–1950.
PubMed Abstract | Publisher Full Text

23. Aristidou A, Lasenby J: *Inverse kinematics: a review of existing techniques and introduction of a new fast iterative solver.* 2009.

24. Pavlakos G, Choutas V, Ghorbani N, *et al.*: **Expressive body capture: 3D hands, face, and body from a single image.** *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2019.

25. Bogo F, Kanazawa A, Lassner C, *et al.*: **Keep it smpl: Automatic estimation of 3d human pose and shape from a single image.** *European Conference on Computer Vision (ECCV).* 2016.

26. Ghorbani N, *et al.*: **human_body_prior: VPoser and optimization-based body priors (software).** 2019.
Reference Source

27. Kanazawa A, Black MJ, Jacobs DW, *et al.*: **End-to-end recovery of human shape and pose.** *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2018.

28. Goel S, Pavlakos G, Rajasegaran J, *et al.*: **Humans in 4d: Reconstructing and tracking humans with transformers.** *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).* 2023.

29. Baradel F, Armando M, Galaaoui S, *et al.*: **Multi-hmr: Multi-person whole-body human mesh recovery in a single shot.** *European Conference on Computer Vision (ECCV).* 2024.
Publisher Full Text

30. Huang Y, Kaufmann M, Aksan E, *et al.*: **Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time.** *ACM Transactions on Graphics (TOG).* 2018; **37**(6): 1–15.

31. Yi X, Zhou Y, Feng X: **Transpose: Real-time 3d human translation and pose estimation with six inertial sensors.** *ACM Transactions On Graphics (TOG).* 2021; **40**(4): 1–13.

32. Yan S, Xiong Y, Lin D: **Spatial temporal graph convolutional networks for skeleton-based action recognition.** *AAAI Conference on Artificial Intelligence (AAAI).* 2018; **32**.
Publisher Full Text

33. Zhao L, Peng X, Yu T, *et al.*: **Semantic graph convolutional networks for 3d human pose regression.** *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 2019.

34. Veličković P, Cucurull G, Casanova A, *et al.*: **Graph attention networks.** *International Conference on Learning Representations (ICLR).* 2018.

35. Huynh DQ: **Metrics for 3D rotations: Comparison and analysis.** *Journal of Mathematical Imaging and Vision.* 2009; **35**(2): 155–164.
Publisher Full Text

36. Umeyama S: **Least-squares estimation of transformation parameters between two point patterns.** *IEEE Trans. Pattern Anal. Mach. Intell.* 2002; **13**(4): 376–380.
Publisher Full Text

37. Mahmood N, Ghorbani N, Troje NF, *et al.*: **Amass: Archive of motion capture as surface shapes.** *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).* 2019.

38. Wang S, *et al.*: **joints2smpl: SMPL-like optimization from 3d joints (software).** 2020.
Reference Source

39. Song H, *et al.*: **Quater-gcn: Enhancing 3D human pose estimation with orientation and semi-supervised training.** 2024. arXiv: 2404.19279

40. Coraggio M, Grotta A, di Bernardo M, *et al.*: **Embodied social experiences in hybrid shared spaces.** 2023.

41. Akada H, Wang J, Shimada S, *et al.*: **Unrealego: A new dataset for robust egocentric 3d human motion capture.** *European Conference on Computer Vision.* Springer; 2022; 1–17.
Publisher Full Text

42. Khan MSU, Stricker D: **Simspine: A biomechanics-aware simulation framework for 3d spine motion annotation and benchmarking.** *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* June 2026; 21176–21187.

43. Taetz B, Lorenz M, Miezal M, *et al.*: **Jointtracker: Real-time inertial kinematic chain tracking with joint position estimation.** *Open Research Europe.* 2025; **4**: 33.
PubMed Abstract | Publisher Full Text | Free Full Text

44. Khan MSU, Wang C-Y: Ik-gat model for neural inverse kinematics and unreal engine ik (ue-ik). dataset. *Zenodo.* July 2026.
Publisher Full Text

Open Peer Review

Current Peer Review Status: ?

Version 1

Reviewer Report 16 September 2026

https://doi.org/10.21956/openreseurope.26439.r80258

© **2026 Pan Y.** This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

?

Yushan Pan
Xi'an Jiaotong Liverpool University, Suzhou, China

The paper proposes **IK-GAT**, a lightweight graph-attention network that reconstructs full-body joint orientations from 3D joint positions through a single forward pass. Overall, the work is valid and scientifically sound. However, I still have the following concerns:

1.In the Abstract, I suggest that the authors assign corresponding numerical labels to the identified problems to be addressed. When presenting the contributions, the authors should explicitly map each contribution to the corresponding problem. This would make the problem–contribution relationship clearer and improve the overall clarity of the manuscript.
2.The Introduction has a similar issue. The problem definitions throughout the manuscript should be kept as consistent as possible. In addition, the proposed innovations should be clearly aligned with the specific problems they are intended to address, so as to ensure the logical coherence of the manuscript.
3.The resolution of all figures should be improved. Some figures remain blurry when enlarged. I recommend using figures with a resolution of at least 300 dpi to ensure sufficient visual quality.
4.The Related Work section should include more discussion of recent mainstream methods. The current discussion of existing approaches is relatively limited and lacks sufficient critical analysis.
5.The overall organization of the Method section should be improved. I suggest adding a brief overview at the beginning of this section to introduce the main components of the proposed method and clarify how they are connected. In addition, a dedicated workflow diagram would be helpful to illustrate the complete information flow through the framework, from the input joint positions to the intermediate representations and final rotation outputs.
6.The distal refinement branch is an actual component of the proposed method, but it lacks a clear problem motivation and problem–module mapping.
7.The Experiments section has a similar issue regarding problem definition. The four questions used to organize the experiments should be kept as consistent as possible with the problems defined earlier in the manuscript, or the authors should clearly establish the correspondence between these experimental questions and the previously identified problems.
8.The authors provide the hyperparameters in the Experiments section, but the specific selection

process should also be provided.
9.One of the core problems that this work aims to address is the sensitivity of classical IK methods to noisy inputs. Therefore, I suggest adding experiments with different levels of noise injected into the 3D joint positions to verify whether this problem is indeed effectively addressed by the proposed method.

Is the work clearly and accurately presented and does it cite the current literature?
Partly

Is the study design appropriate and does the work have academic merit?
Partly

Are sufficient details of methods and analysis provided to allow replication by others?
Partly

If applicable, is the statistical analysis and its interpretation appropriate?
Partly

Are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions drawn adequately supported by the results?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Affective Computing, Multimodal Perception, Human–Computer Interaction, Human–Robot Interaction, Machine Learning, Graph Neural Networks, and Human Motion Analysis

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.